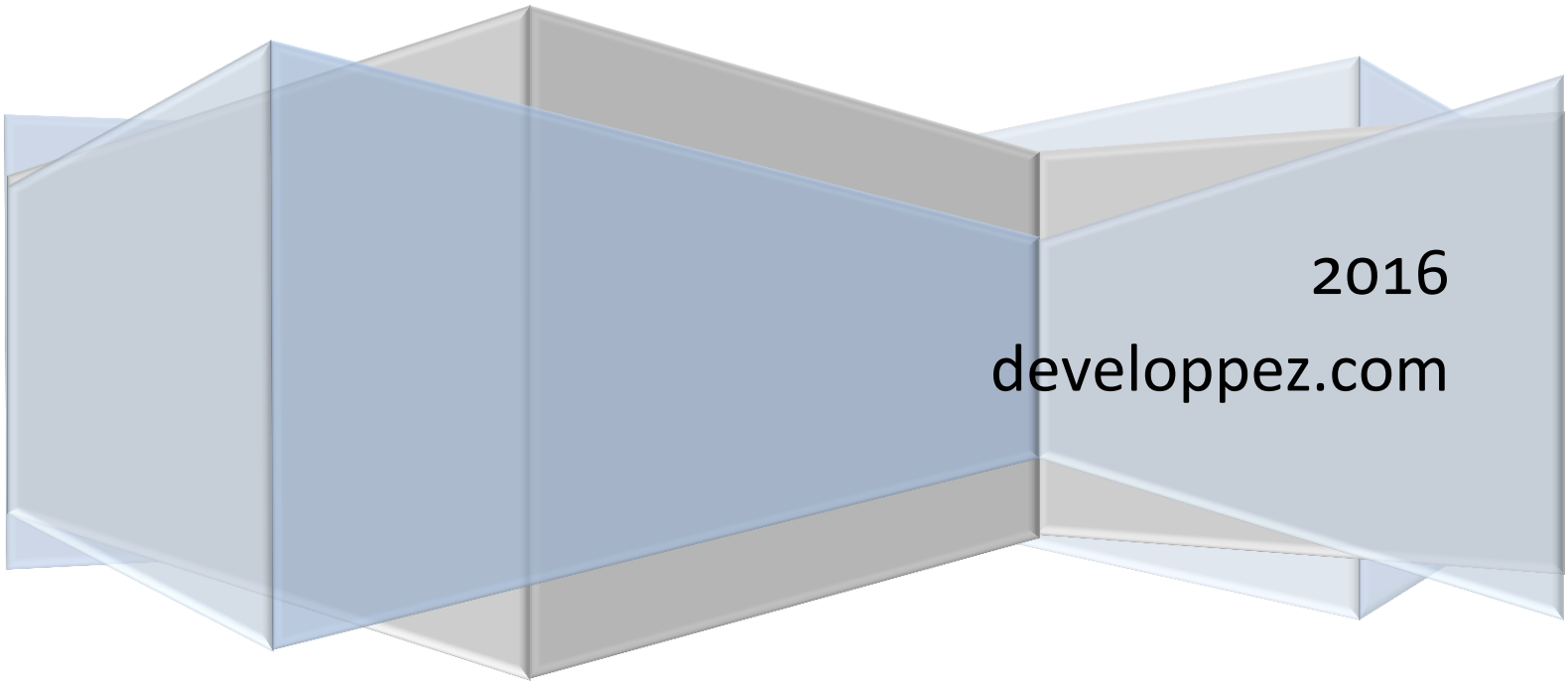
GVSoft

PASHeader 0.97

Getting started

Gilles Vasseur



2016

developpez.com

PASHeader 0.97

Introduction

To be useful and effective, headers of source files must be informative, clear and a little bit aesthetic. **PASHeader** is just that kind of software that allows you to create and modify headers for Pascal, Delphi and Lazarus files. It specially demonstrates its usefulness with projects of some complexity that would otherwise require unsafe manual updates.

A cycle of work with **PASHeader** only requires three steps:

- choose files you want to modify ;
- define or load a new header ;
- start processing files of the first step.

PASHeader provides the following advanced features:

- individual processing or batch files;
- complete editor to define headers;
- automatic extraction of files used by a project or package;
- automatic cleaning of existing headers;
- file preview;
- customizable inclusion/exclusion for each file, as well as for cleaning treatment;
- definition of a new header from a template or manually;
- automatic extraction of info version from a Delphi project or Lazarus project or package;
- publishing simplification tools with manual, semi-automatic or automatic pre-filling templates (15 predefined keywords and 4 copyright notices);
- validity check tool of headers;
- creation of general templates for easy reuse;
- smart update of a model from supplied and/or extracted data;
- preview of the execution to prevent errors (with warnings if necessary);
- execution with backup of changed files;
- saving and loading projects if needed;
- options page saved for a better user experience;
- English or French easy-to-switch translation of text in **PASHeader** (expandable with PO files in the *languages* directory);
- full source (100% Lazarus Pascal under GNU GPL license) with documentation and predefined headers (*templates* directory);
- tested with Windows 10 (64 bits), Windows 8.1 (32 bits) and Ubuntu 14.04.

PASHeader was produced for the 2016 Pascal challenge created by the Pascal/Delphi/Lazarus group of the French website http://developpez.com. **PASHeader** and all files in its directory (including this document) are under GNU GPL license. You can get a copy of this license at: http://www.gnu.org/copyleft/gpl.html.

Enjoy!

Gilles Vasseur

Contact: gillesvasseur58@gmail.com

Personal website: http://lettresenstock.org/

Work on developpez.com: http://gilles-vasseur.developpez.com/

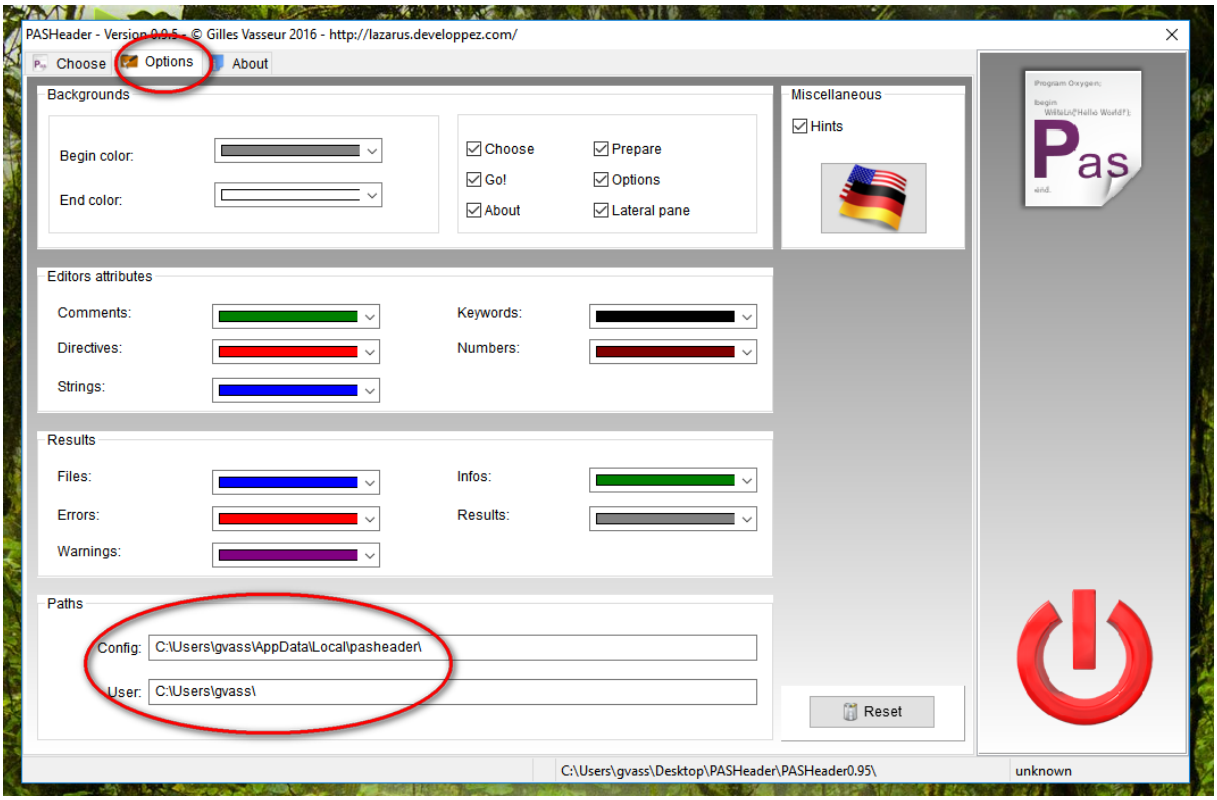
Movie tutorial of **PASHeader** (French): https://www.youtube.com/watch?v=T7otZylbuX0

How to install

Installing PASHeader

To use **PASHeader**, you just have to copy its directory wherever you want (or can!). For instance, you can copy files into your usual workplace, generally called the « *documents* » directory. **PASHeader** will anyway create its own directories to save configuration files, templates and projects.

Configuration files and templates are saved in the same place. Projects have their own directory. You can have a look at their locations thanks to the "*options*" panel:

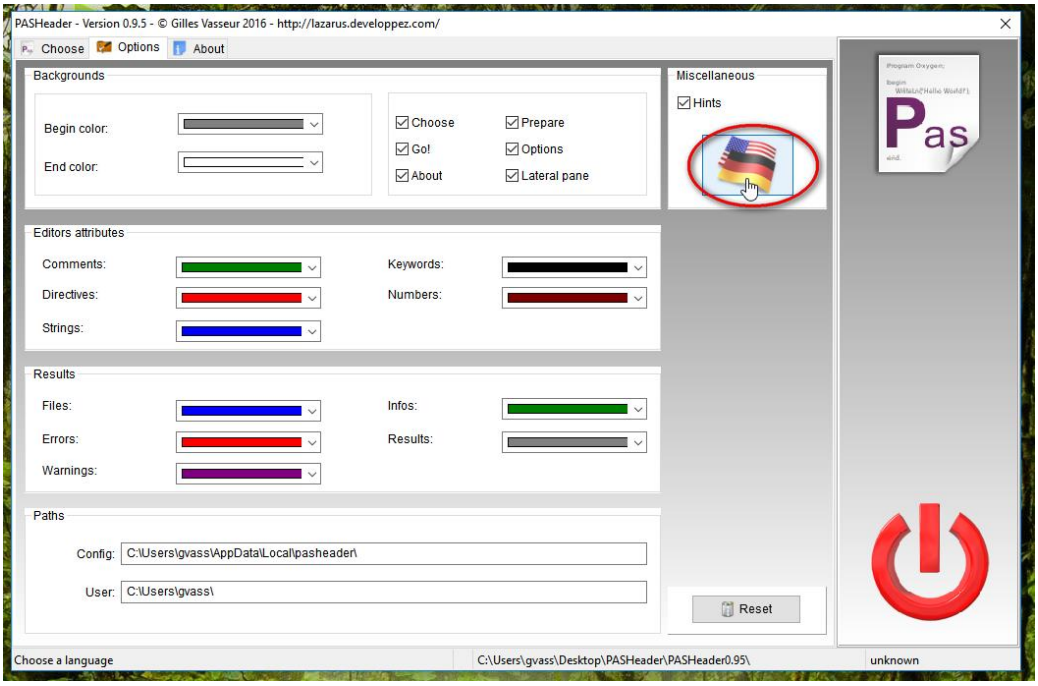


You may only modify the « *user* » directory while the "*configuration*" directory is defined by the OS.

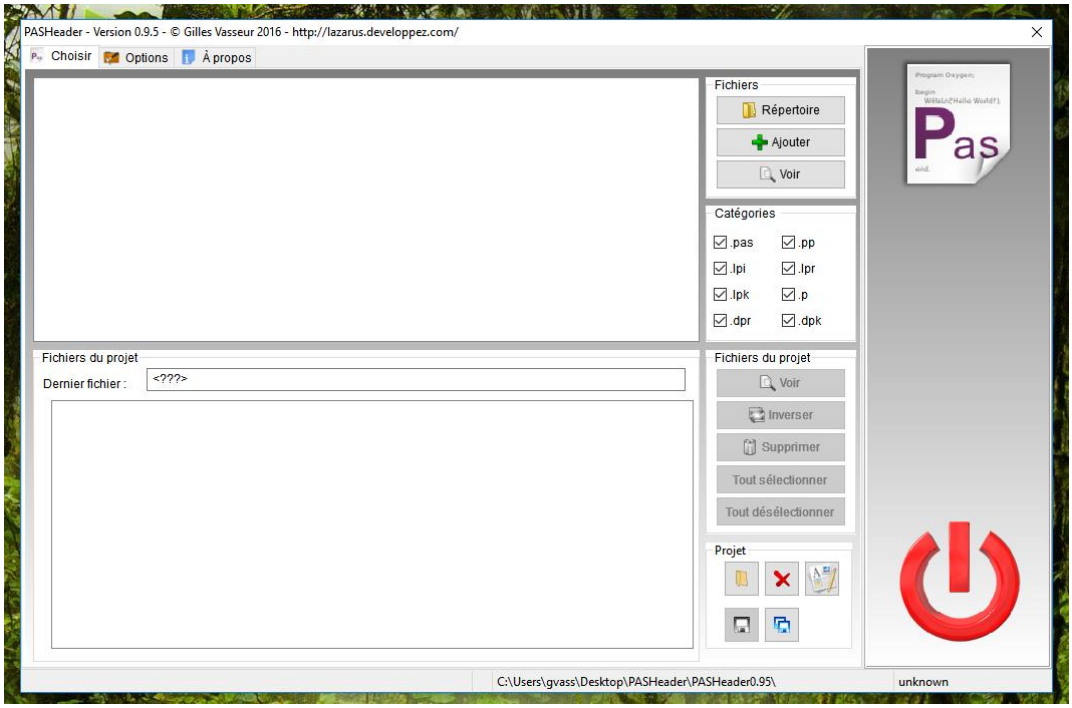
If you get error messages when saving projects or templates, or when closing **PASHeader**, you certainly have no permission to write to the current "*user*" directory. Please, make a better choice!

Working with another language than English

On first use, **PASHeader** works in English. French people have to click the flag in the "*options*" panel to get an immediate translation of all messages to their proper language.



Here is a snapshot of the French screen:



To get back to English, click again the same flag. **PASHeader** always memorizes your last choice.

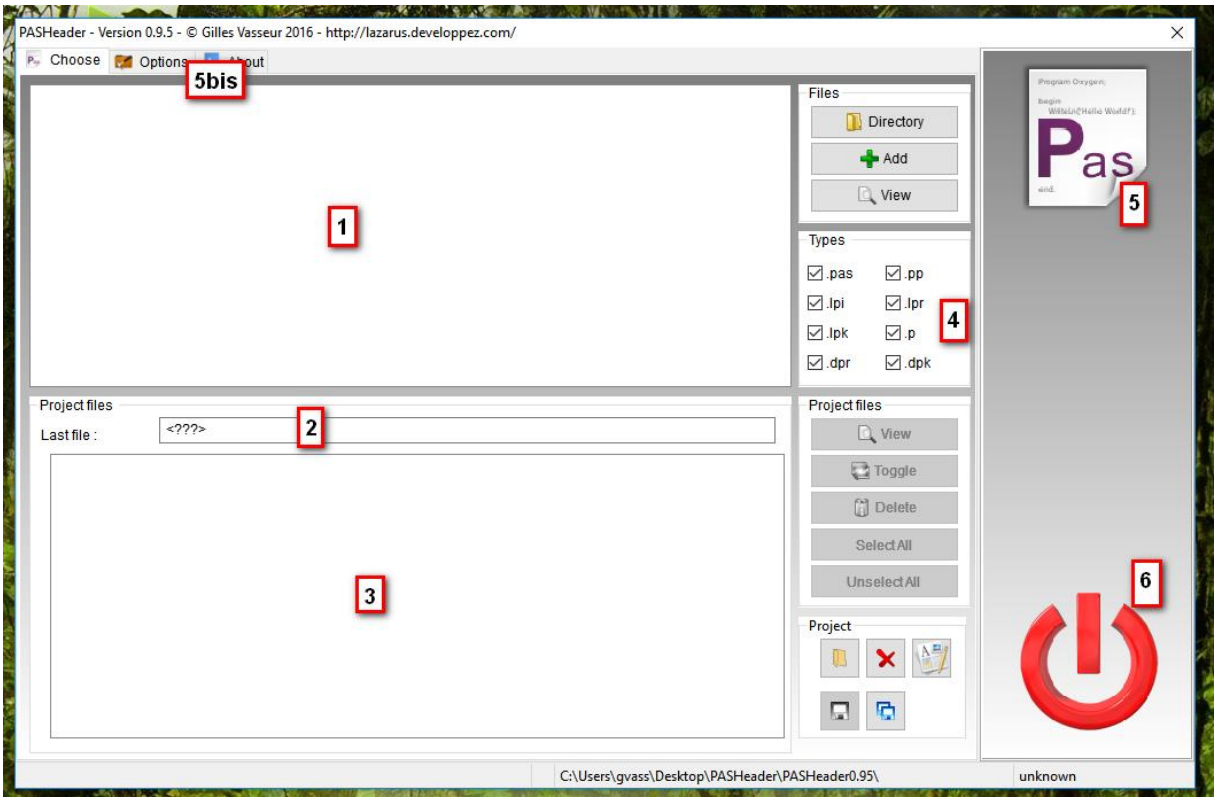
An user-defined header for dummies

The aim of the following tutorial is to help you to learn the basis of **PASHeader**. Working with a single file, it may not be representative of what you really can get with this software!

Startup screen

As already indicated, **PASHeader** works in three steps: choosing files to be modified, setting up a new header and applying changes to files of the first step.

At startup, the main window looks like this:



Area 1 will display the name of files from the directory you will have chosen first. It's empty if this directory doesn't contain any valid files for **PASHeader**.

Area 2 will display the name of the last file added to the list of files to modify. You can drop files here instead of using buttons.

Area 3 will display the name of files ready to be modified.

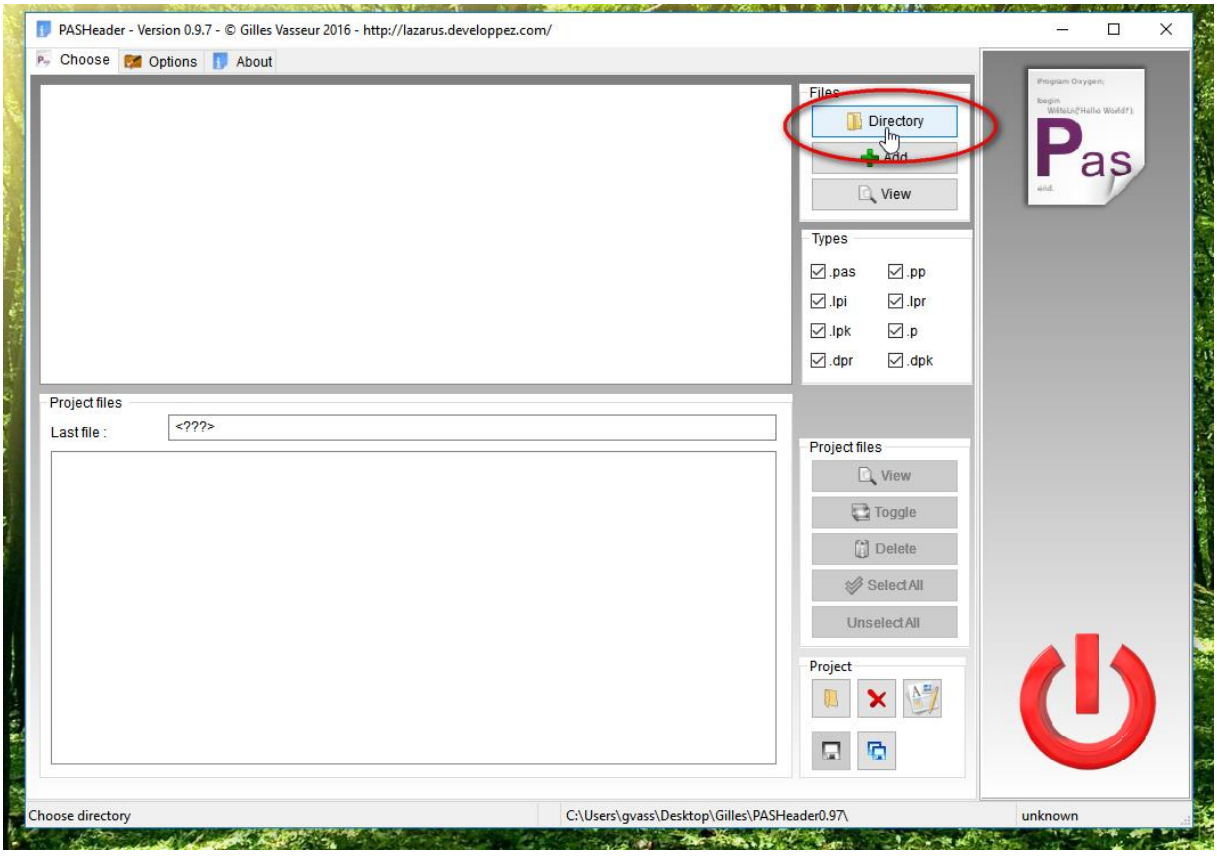
Area 4 consists of checkboxes that indicate what kind of files will be displayed on area 1. It stands above a first zone of grayed buttons indicating no files are selected, as well as a second zone displayed on all actions panels that includes commands for the current project. These commands will be discussed later.

Area 5 consists of large buttons with pictures to quick access to panels of area 5 b. Visibility of these buttons depends on your work: after choosing files to edit, a button will appear to set up the new header. Likewise, after preparing the new header, a third button will allow you to apply changes. These appearances follow the chronological order in three steps mentioned above.

Button 6 allows you to exit **PASHeader**. When the project was changed, you are prompted to save your work before exiting.

How to choose files

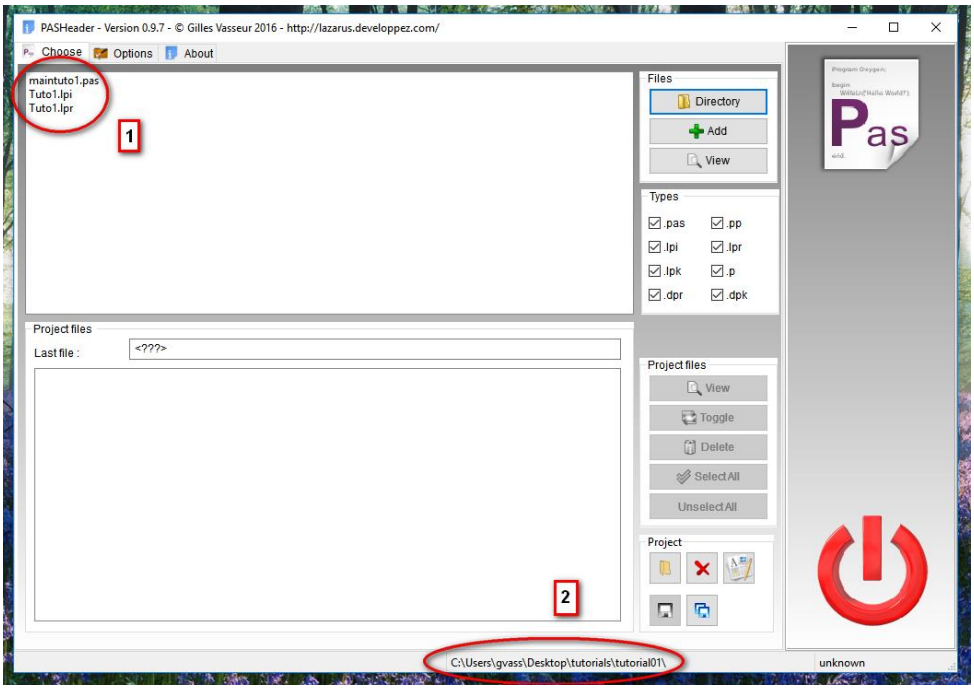
For this first tutorial, you will use a dummy program in the "*tutorials01*" directory of **PASHeader**. Click "*directory*" on the upper right side of the main window and then select the appropriate directory:



Remember that to properly work with files of the tutorial you have to use a folder for which you have reading and writing rights.

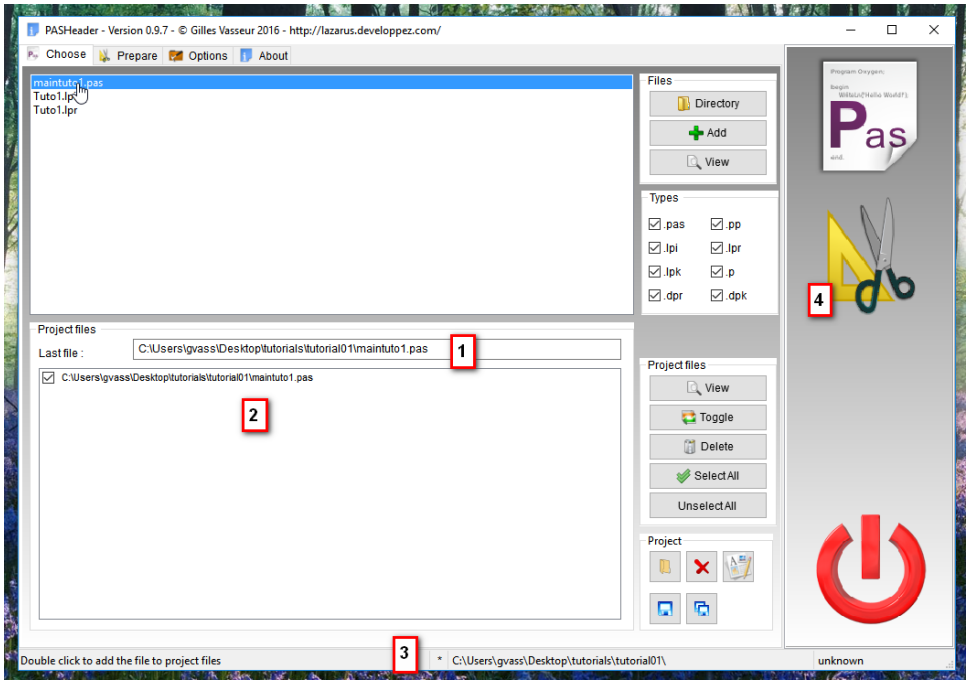
Once you validate this directory clicking "*Ok*", you will find appropriate files listed on the first panel according to filters of area 4.

Here is the main window you should see:



Area **1** displays the names of the three files of the selected directory that can be treated by **PASHeader**. In area **2**, the taskbar has been changed according to the new working directory.

For your first experiment, double-click on "*maintuto1.pas*" to ensure this unit will be added to the edit control labeled "*last file*" and to the panel below it:



The name of the selected file now appears in the edit control 1 and in the list box 2. As it is checked, its header will be cleaned before the insertion of the new header.

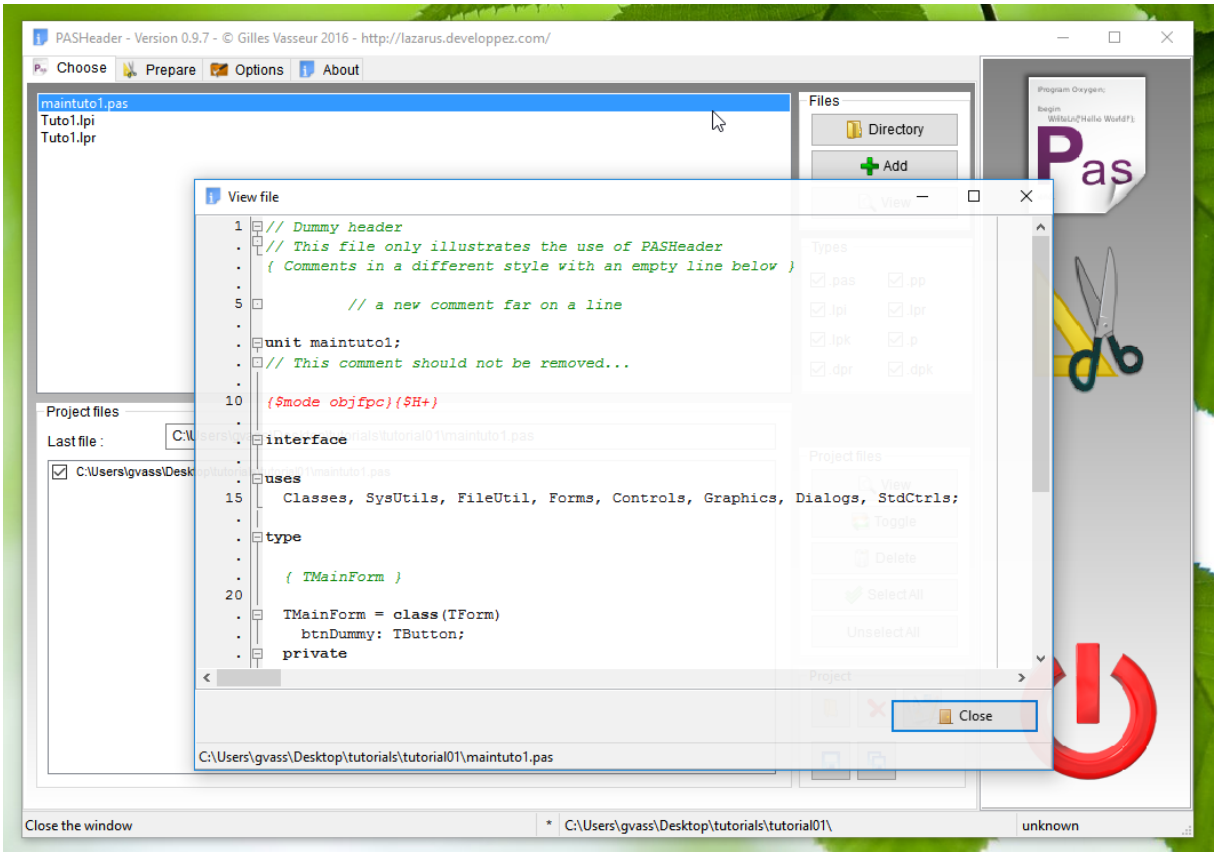
On the taskbar 3, there is an asterisk indicating that the current project has been modified.

Finally, in area 4, there is a new picture that will allow access to the page for creating the new header. It will be your second step.

How to view files

You have three ways to view the contents of files:

- right click the name of the file and choose “view file” in the popup menu;
- click “view” on the right of the panel where you can see the name of the selected file;
- drag the name of the selected file onto the « view » button and drop it there.

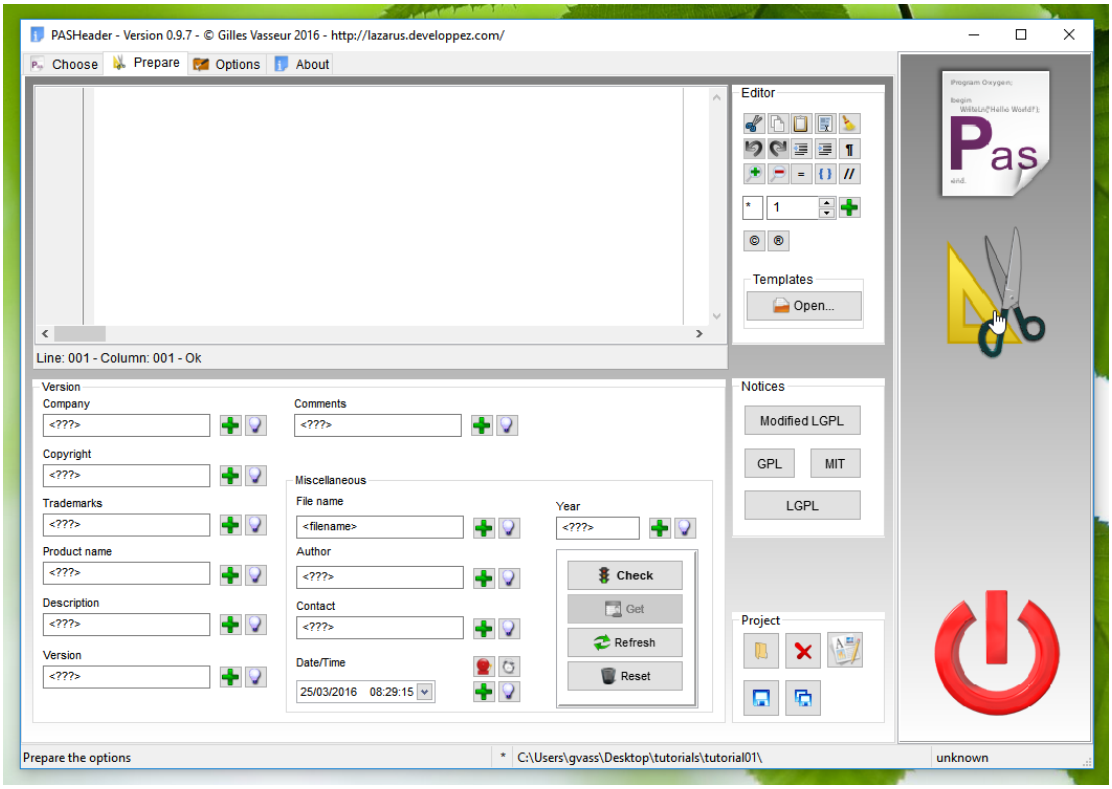


How to design a new header

For this tutorial, you will leave the check because the file has a header that should be removed.

The new header will be quite simple and will only use a few tools of **PASHeader**.

- Click the scissors picture or the "*prepare*" panel to design your new header :



Although headers could be built manually as you would do with an ordinary text editor, some features of **PASHeader** will be very useful if you want to avoid a strike surplus. In addition, a tutorial is always the chance of discovery, isn't it?

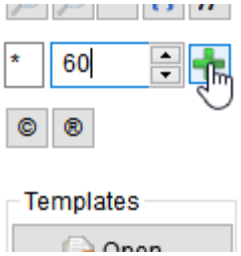
Start by filling the following version information:

- copyright: "© I me myself";
- description: "doesn't do anything".

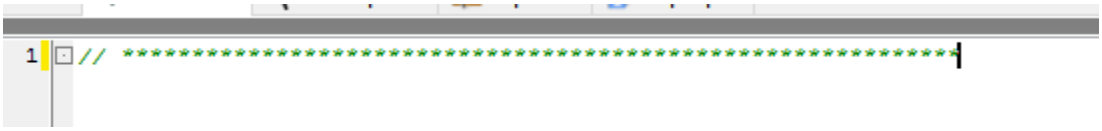
Here is what you should obtain:

You will now create the new header.

- Enter **60** in the spin edit control on the "editor" panel and click the "**+**" button. 60 stars are added to the editor:

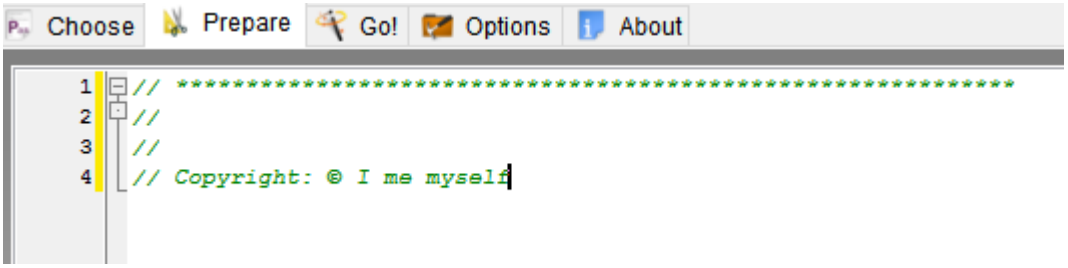


- Click the // button to insert a new comment sign:



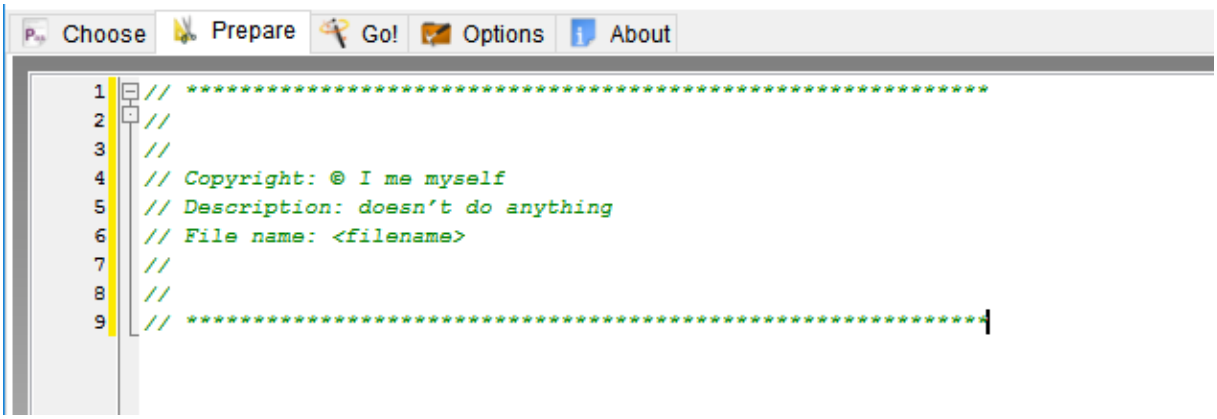
Syntax highlighting is the one used by the Lazarus default editor.

- Add a line break, by clicking either "*enter*" or ¶.
- Click once again //, ¶, //, and then ¶ to have two new lines of comments.
- Click on the right of the "*copyright*" zone and then on // to obtain next screen:



Most key combinations of the Lazarus editor can be used: you therefore will have little habits to change.

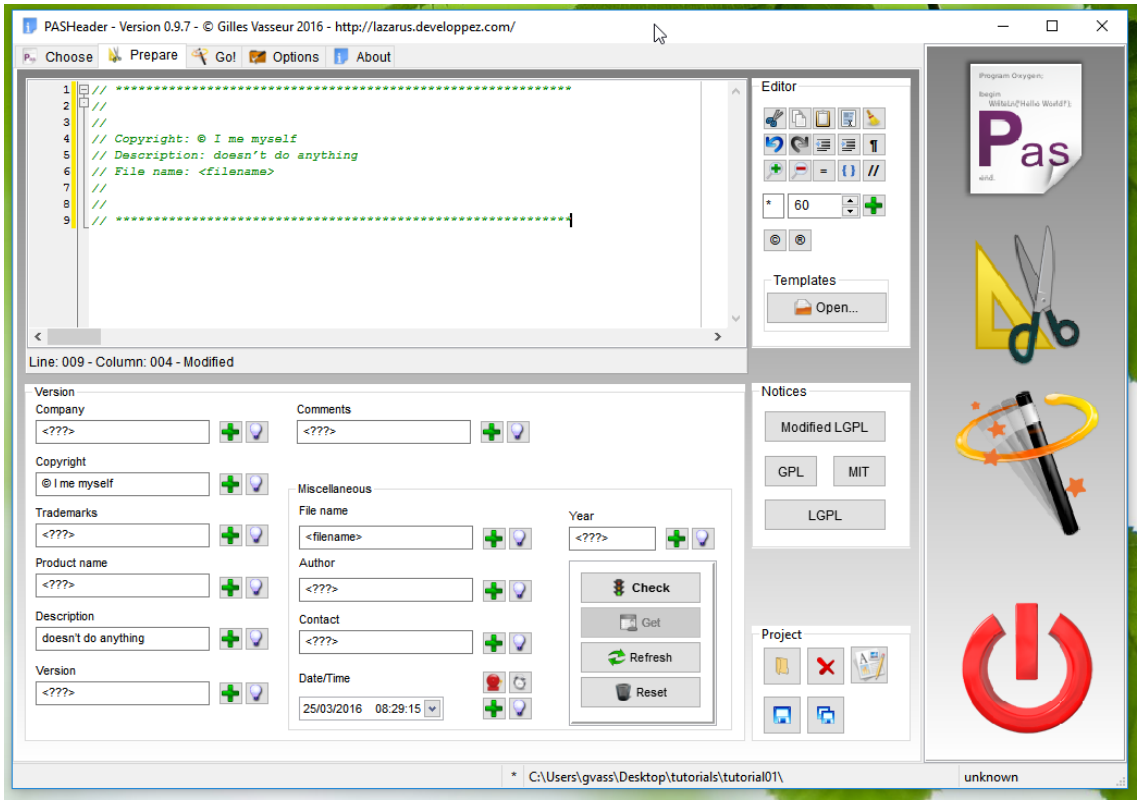
- Fill the screen in the same way using the button  or keyboard for the following:



The only new feature is the peculiar word "*<filename>*". This is a keyword that is to say a word **PASHeader** can identify and handle. "*<filename>*" means that the name of the actual unit file will replace it when processing.

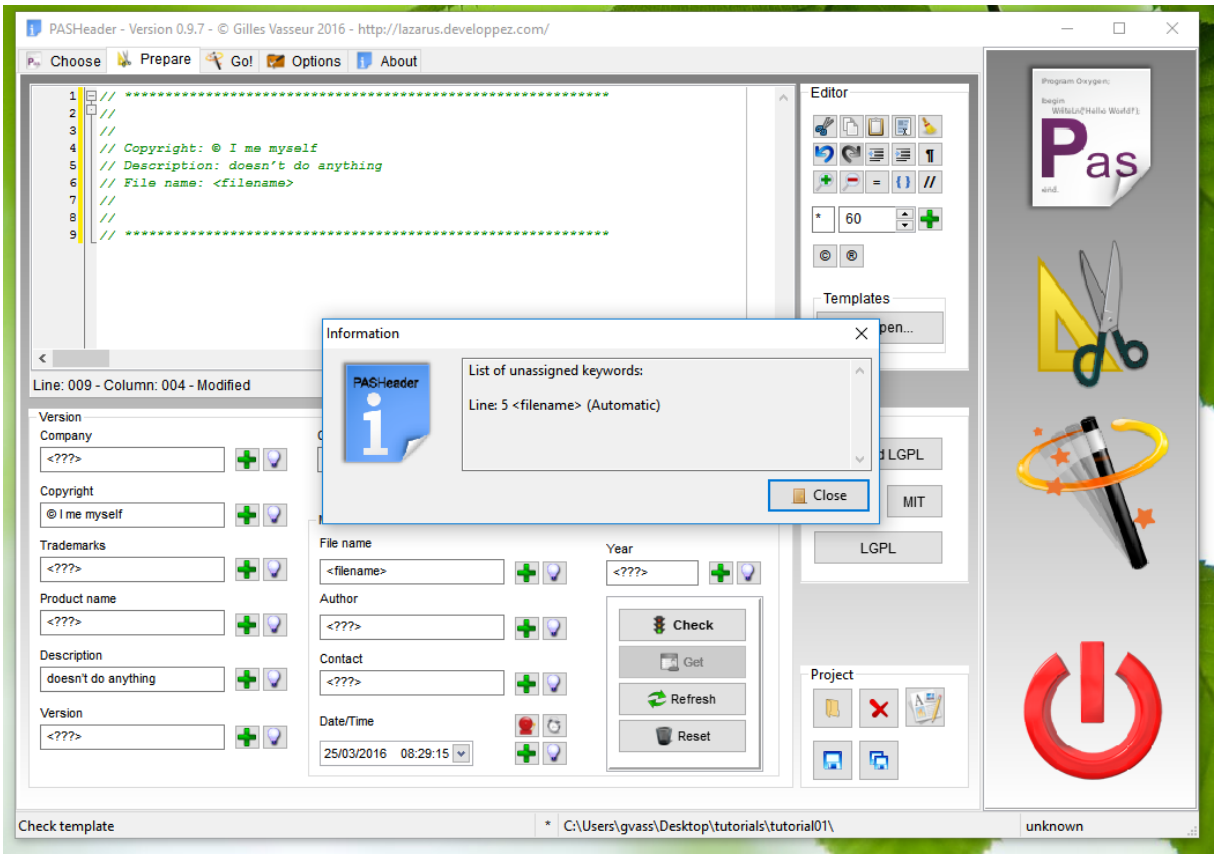
Obviously, in the case of a single file, the simple file name could have been used, but batch imposes such a generic word.

The final screen of this second step should match this:



If the characters in the editor seem too small or too large, or if you want to preview your work, it is possible to adjust the display, either by clicking one of the three buttons , or holding down the **Ctrl** key while moving the mouse wheel.

You can now check the validity of your header by clicking "*check*". A window instantly provides an assessment of your virtual conversion header: for the example treated, it mentions that the <*filename*> field was left blank, but specifies it will be automatically populated.



How to apply changes

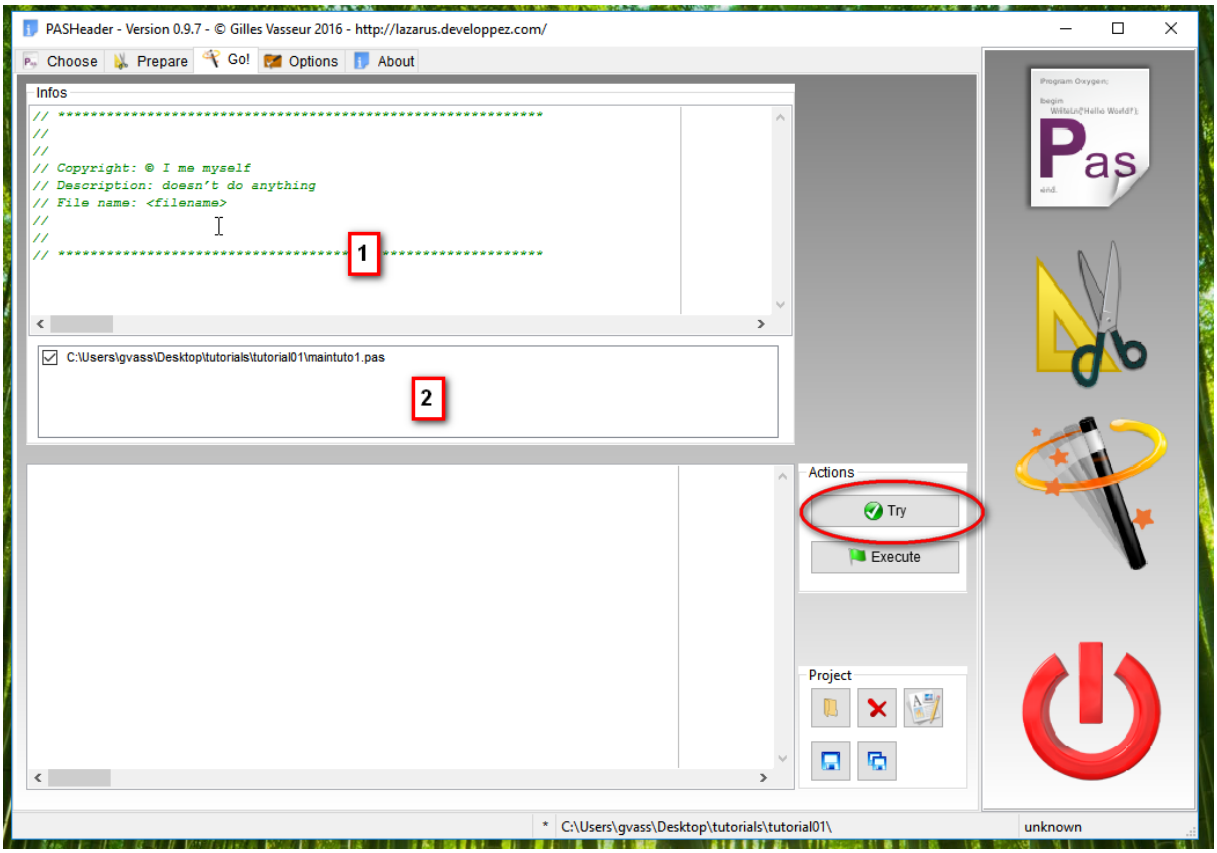
You have chosen a file and set its header: you can now apply it! You may have noticed a new and final picture that represents a wand appeared in the right band of the software. With this wand you can display the last panel.

Testing a script

See you on the page of the script by clicking on the magic wand or on the panel "*Go*!".

You will notice that it is first possible to test the validity of all your preparation.

Here is the current window:

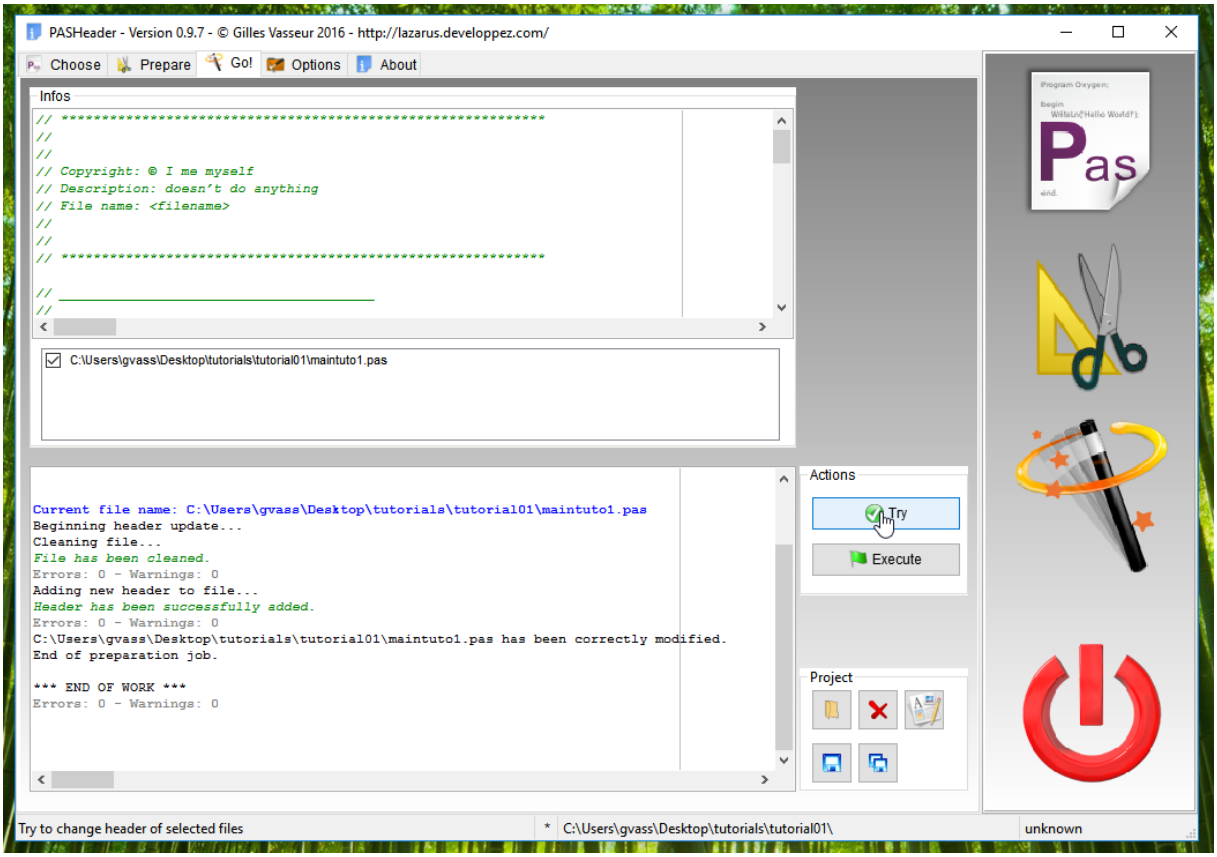


In area **1** is reproduced the header defined above. It is also here that will be inserted the modified files for verification.

Area **2** contains the list of the selected files and the indication by a check of their pre-cleaning that will remove all old headers.

- Click "*try*".

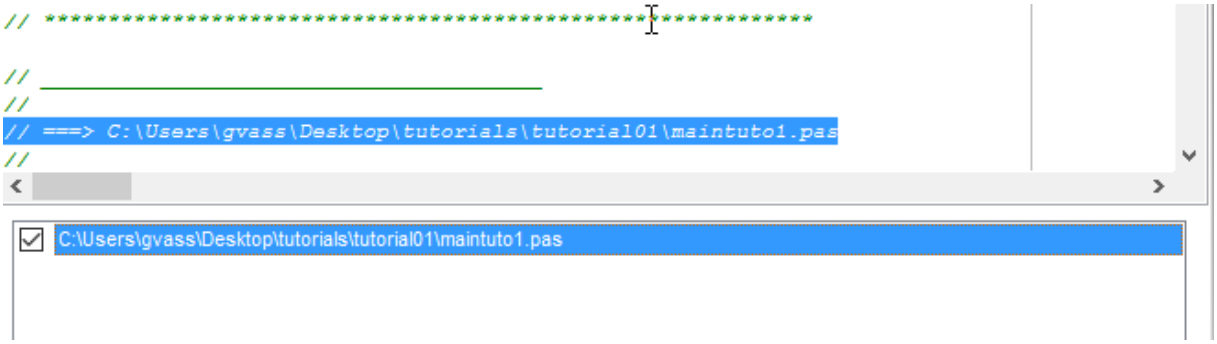
In a split second, the script is executed without affecting the original files. The lower part of the page shows the analysis as it was carried out and reports any warnings and errors.



Although **PASHeader** interrupts its work on error and makes a backup of changed files, it is better to check the script *before* changing actual files. Indeed, recovery is always more difficult than prevention!

If you click the file name preceded by a check mark, the editor "*infos*" leads you to a line that introduces a reproduction of the file as it will be once modified.

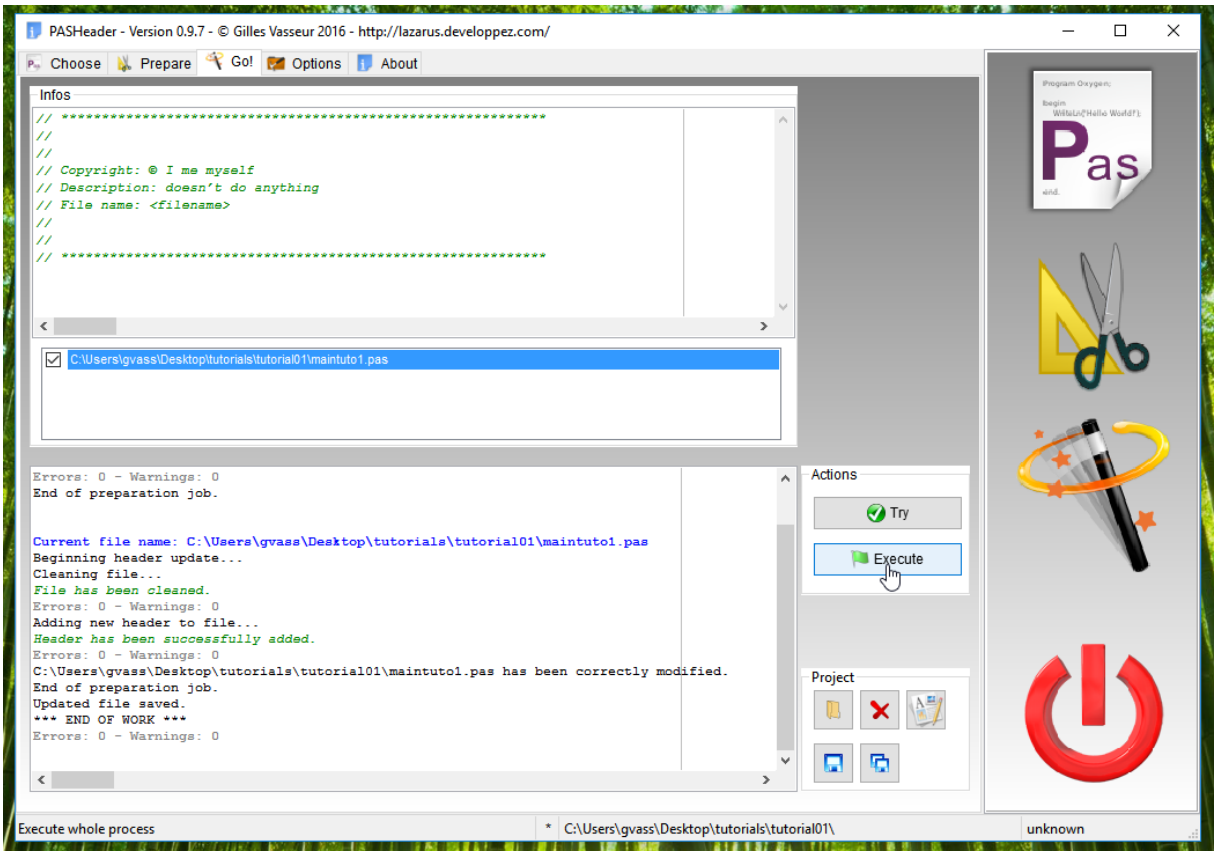
- Make scroll the text in the editor to see changes.



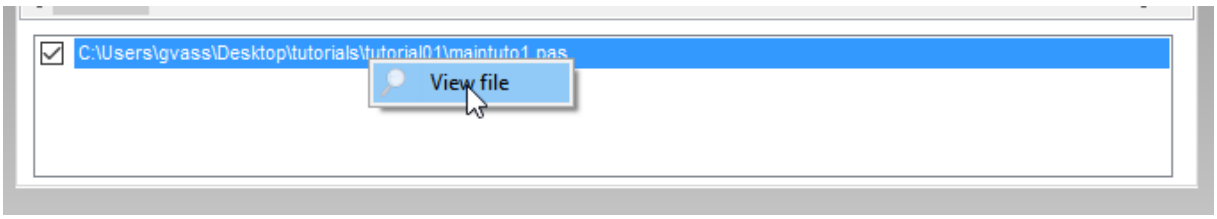
In particular, the old header was removed (and only it!) and the keyword <*filename*> has been replaced by the actual file name.

Running a script

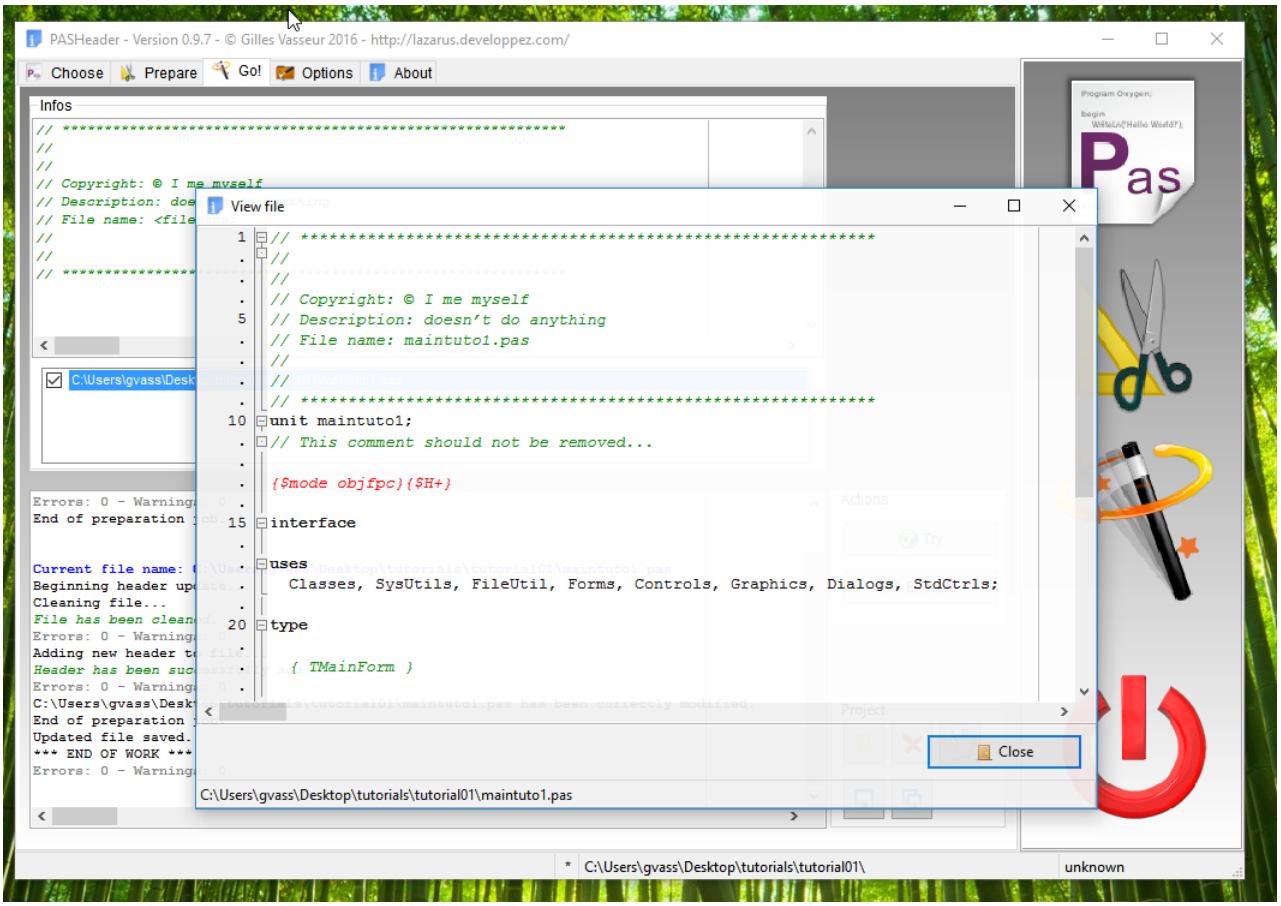
As the script corresponds to your expectations, you can run the final script. Click *"execute"* and look at changes:



The "*infos*" display area only reproduces the predefined header while the lower one shows again a series of lines indicating the actions undertaken and their outcome. By clicking the right mouse button on the file checked, a popup menu allows you to view it as amended:



Here is the updated file:

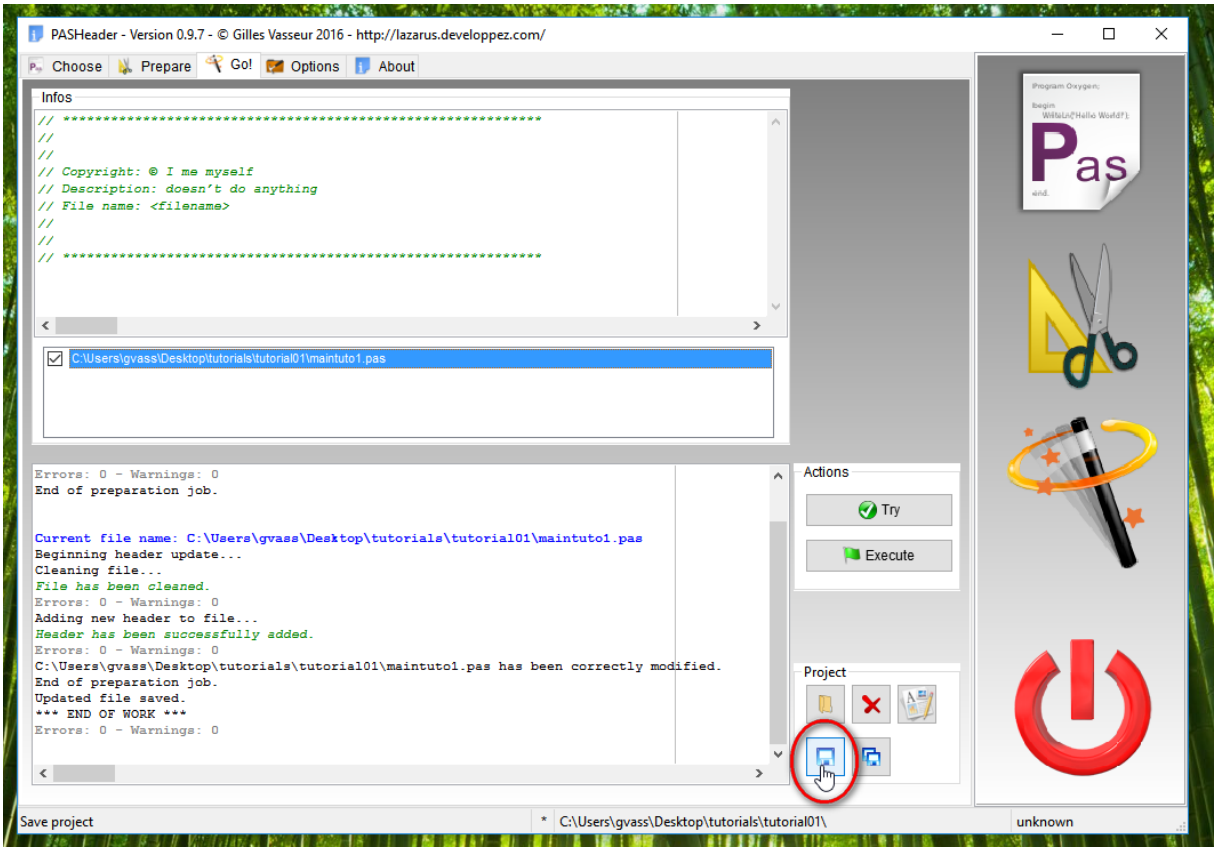


This file is exactly the one you expected!

Saving your project

It often makes sense to save your header project. For example, if you later change the description of your software, if only as the version number, handling will be reduced: you'll only have to load your project, change the targeted information and all selected files will be updated in one click!

- Click the button representing a floppy disk:



A registration window opens at the location provided by the specified path on the "*options*" page.

- Type "*tutorial1*" and save your work.

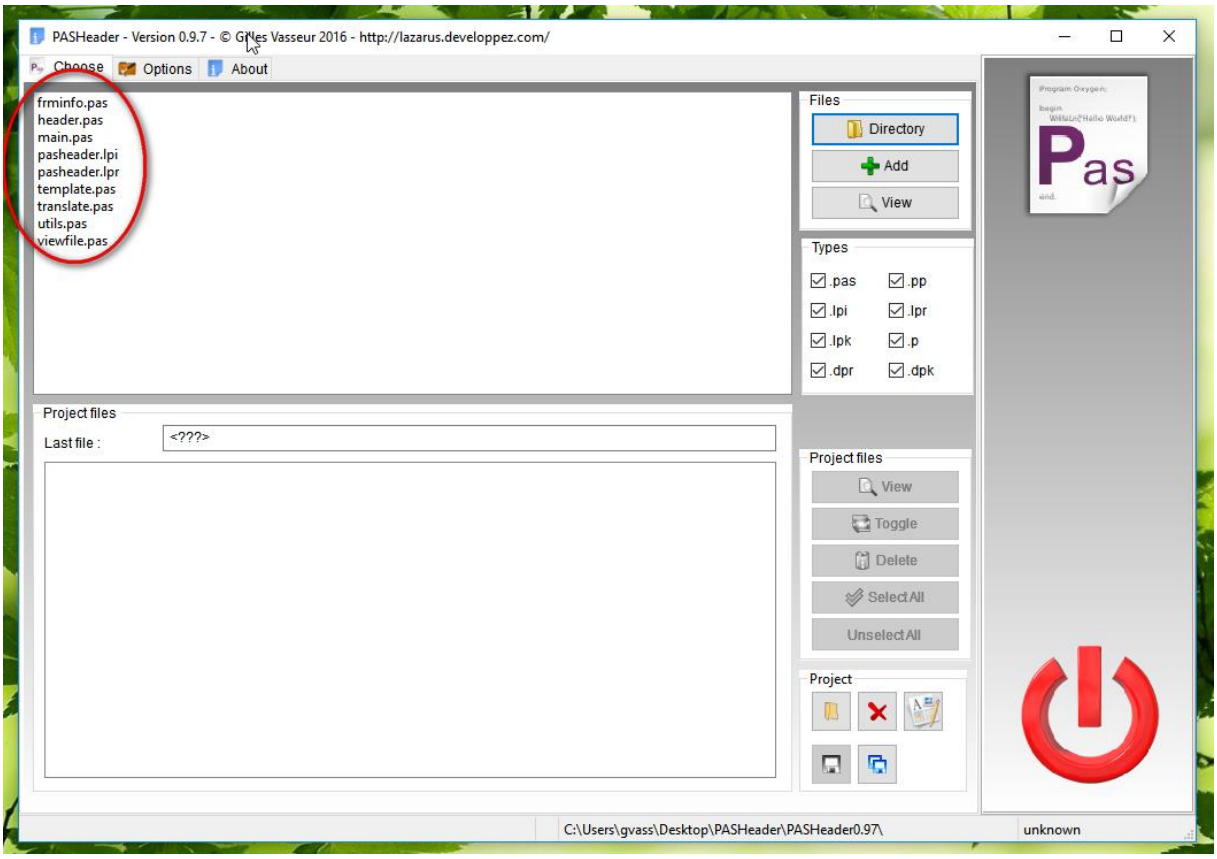
This first tutorial is complete. Thereafter, you will discover that **PASHeader** offers more powerful features ...

How to handle a package or project

This tutorial assumes you have read the basic steps as defined in the first tutorial. After the simple modification of a file, it's time to change an entire project, i.e. the source code of **PASHeader** itself!

How to choose a project/package

From the "*choose*" panel, choose the directory where you stored the source code of **PASHeader**. The new screen that appears should have integrated the software code files:

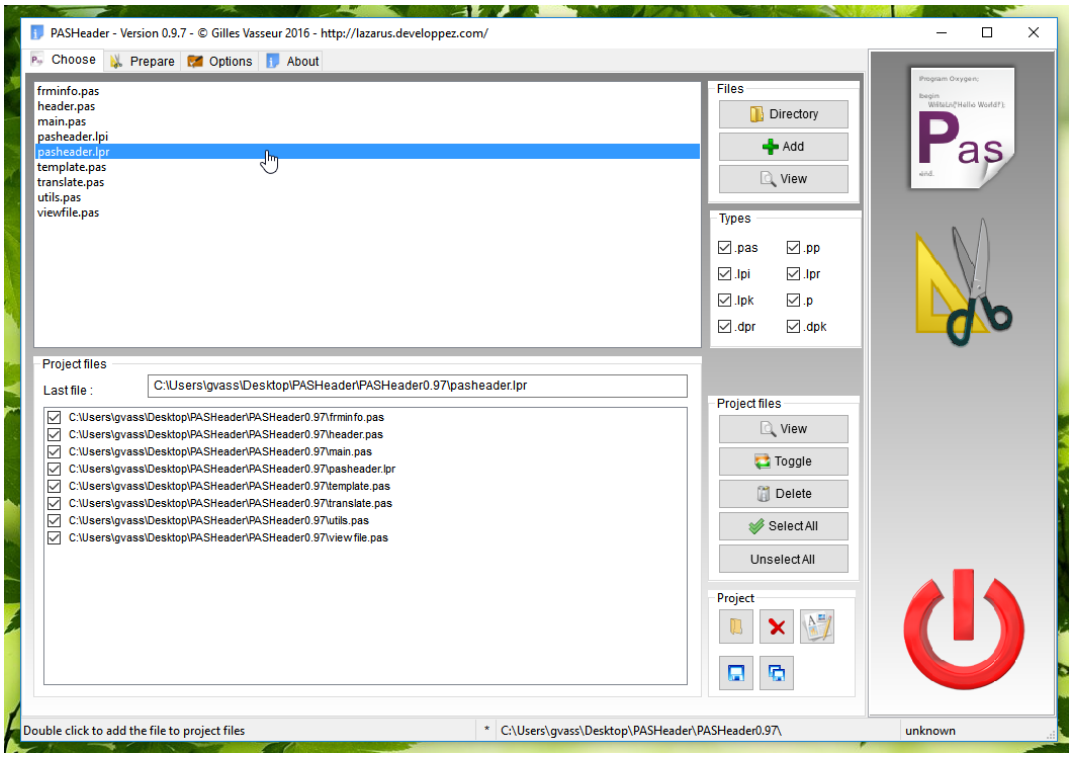


As for the display of a file, three options are available to include an entire project or a package in the area of the files to be treated:

- double-click a file representing the project or package ("*.lpi*"», "*.lpr*", "*.lpk*", "*.dpr*" or "*.dpk*" extension);
- click as before on a project file or package, then click the "*add*" button on the right panel;
- drag the name of the same file to the display line labeled "*last file*" and drop it on there.

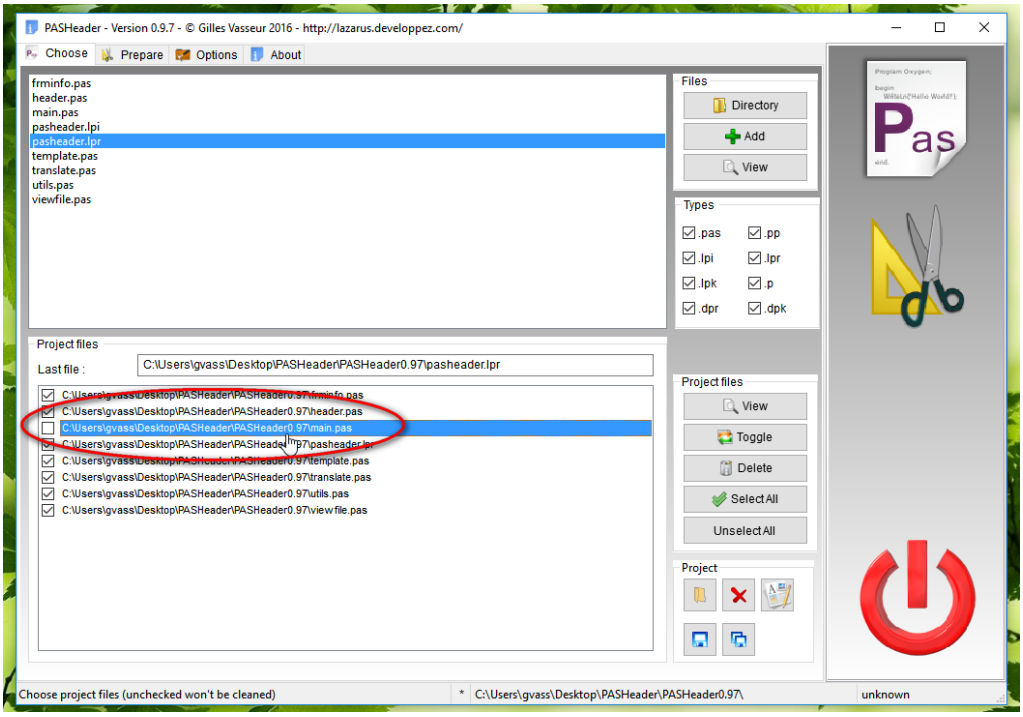
In the current example, you can choose "*pasheader.lpi*" or "*pasheader.lpr*". Once the action performed, all the project files are built into the lower display area of the screen.

Here is the screen that you'll get:



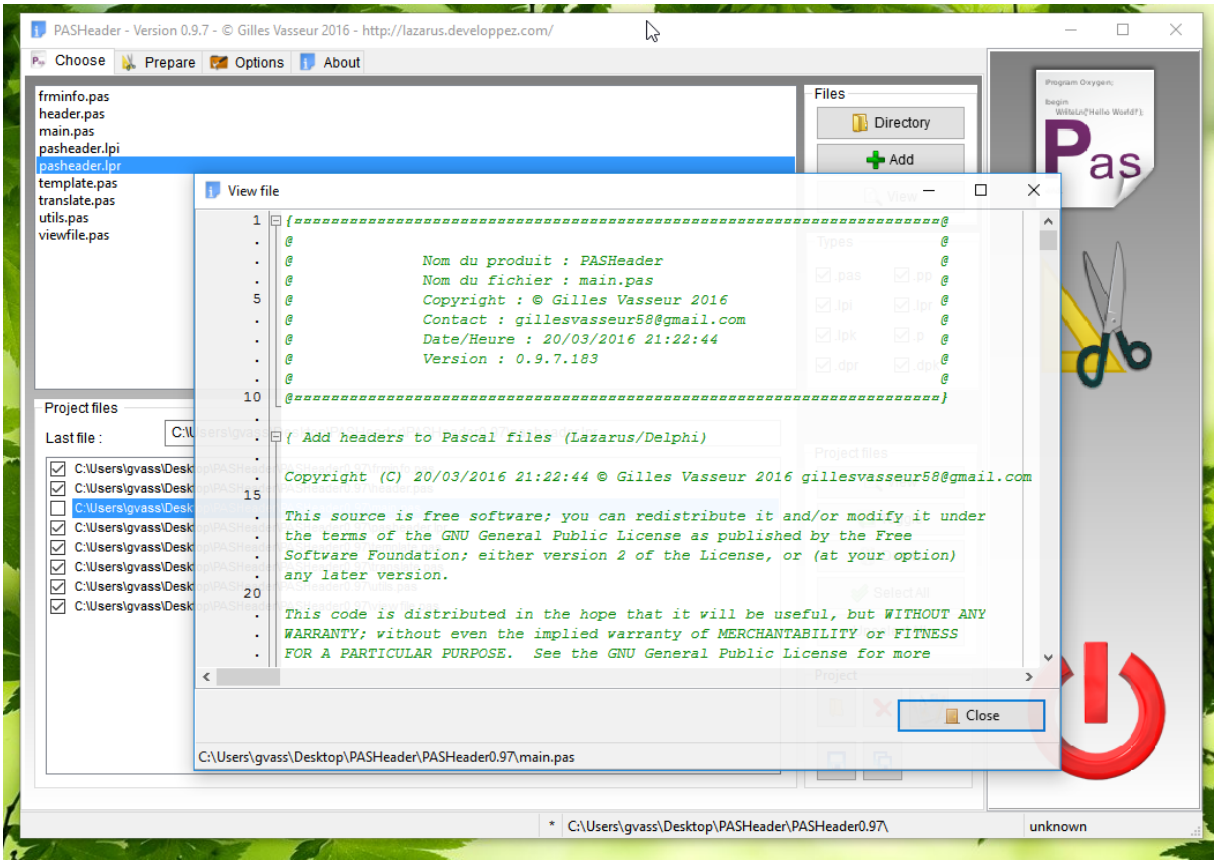
To test that the headers to change can be controlled from the checkmark in front of the file name, you'll clear one, indicating that the current header should not be deleted.

- Uncheck "*main.pas*".



You can view the current headers as in the first tutorial: this will allow you to properly account for changes.

- For example, here is the header of "*frminfo.pas*" file:

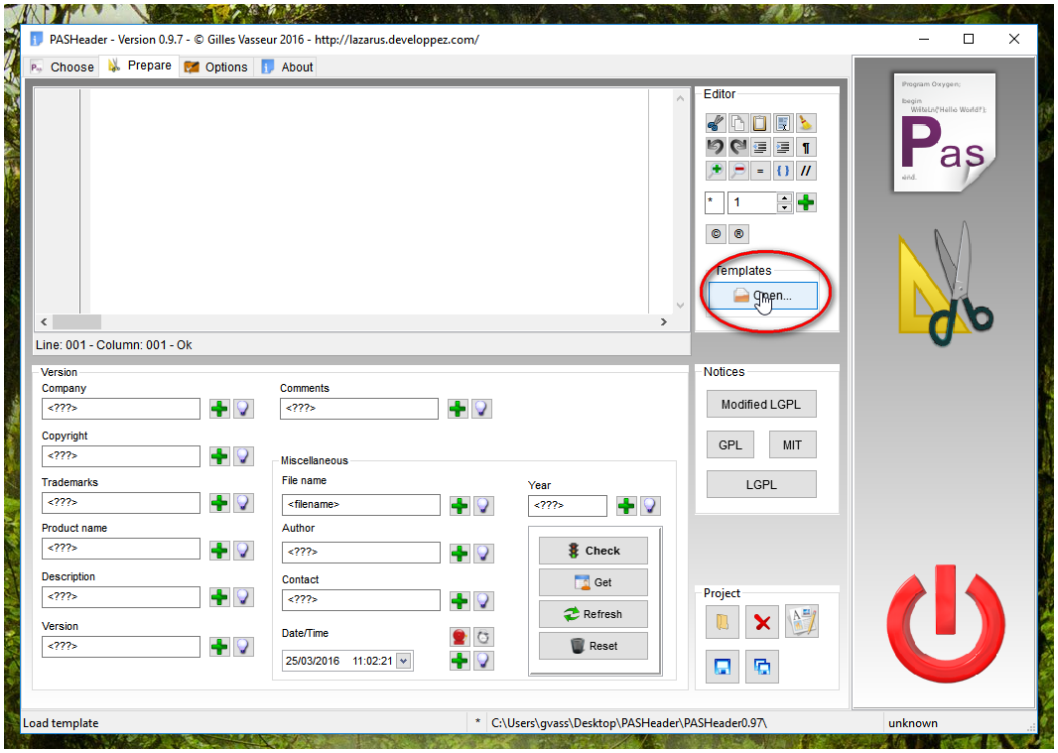


As you can see, current header is written in French : you probably prefer English instead!

How to prepare the new header

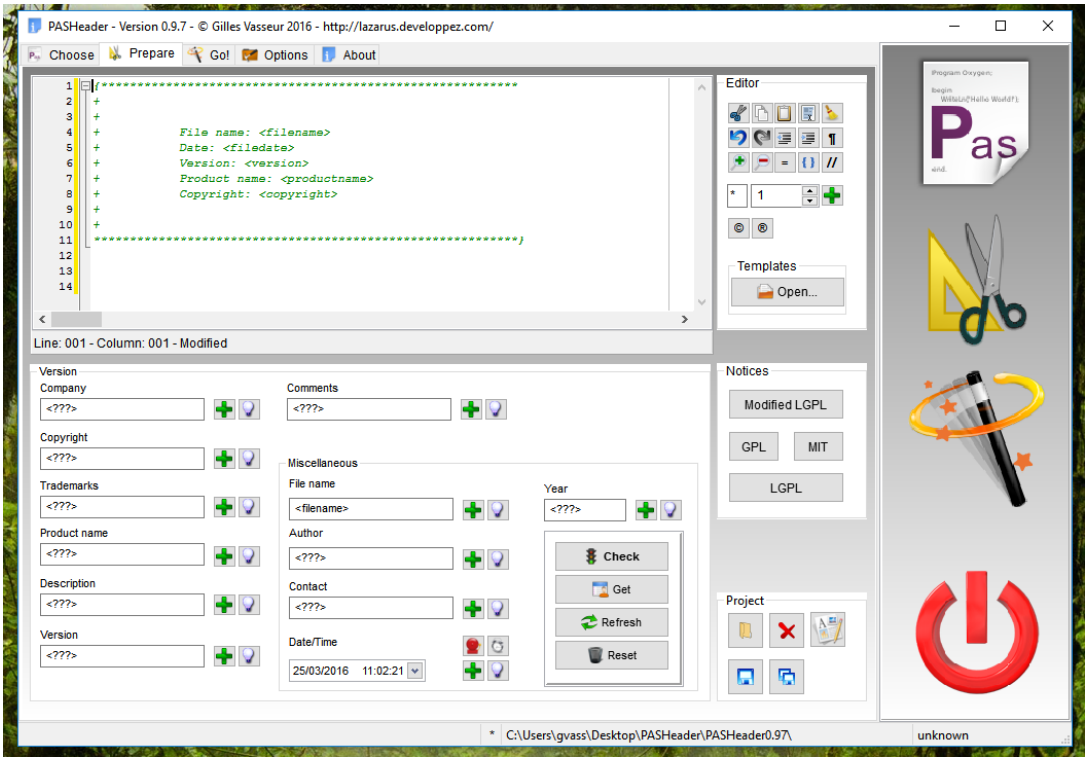
Now go to the "*prepare*" page. You will first choose an appropriate template to base your new header.

- Click "*open*" in the panel "*templates*" to the right of the editor:



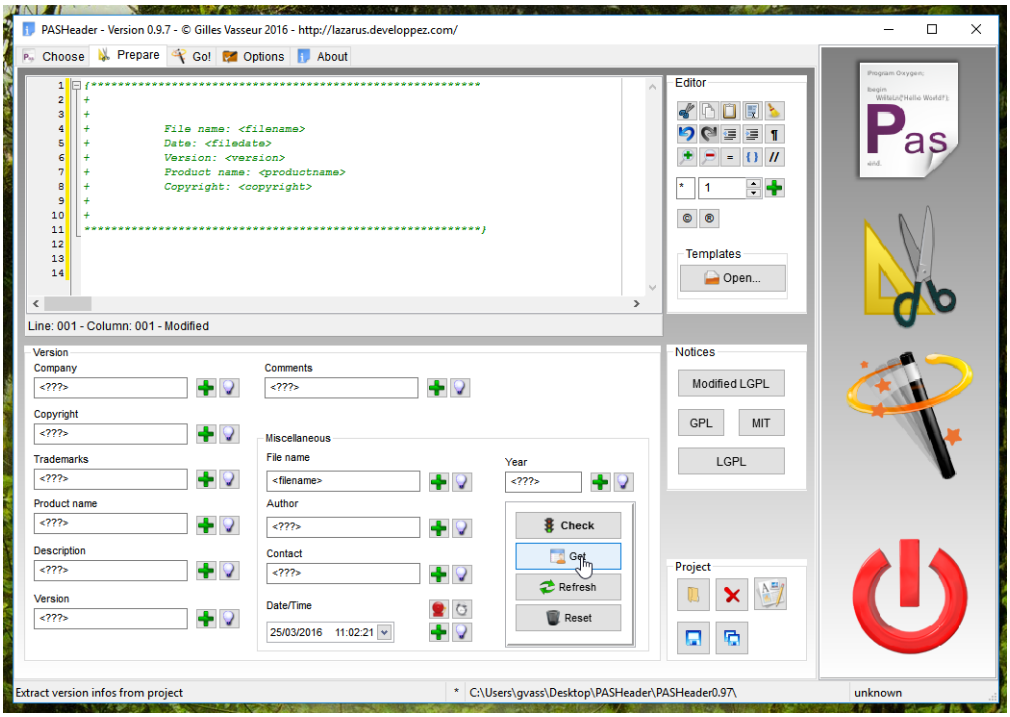
- Then select the template labeled "*simple.tph*" from the *"templates_en"* directory..

The editor is completed by the data recorded for this template:

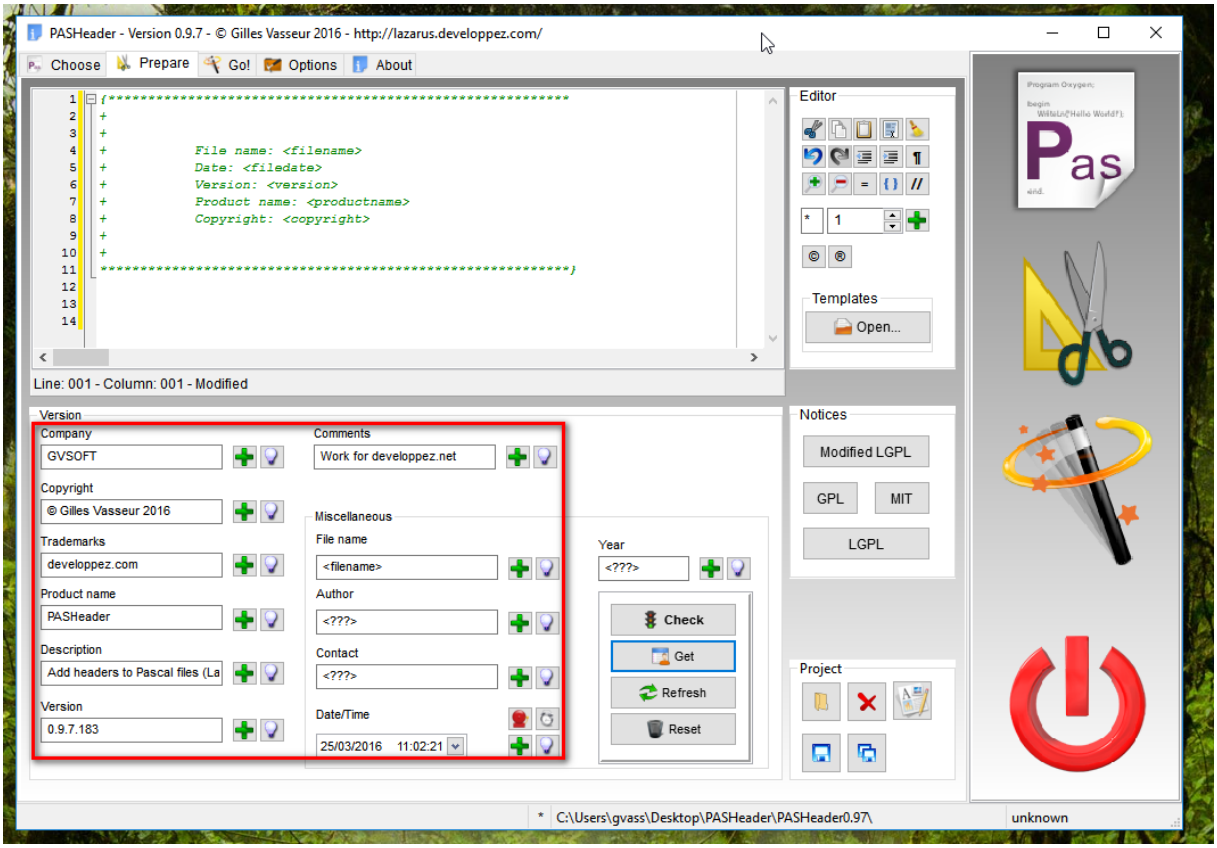


As the name suggests, it's rather basic, but will be suitable for our initiation work!

As the program is a Lazarus program and can include version information, the button labeled "*get*" is enabled:

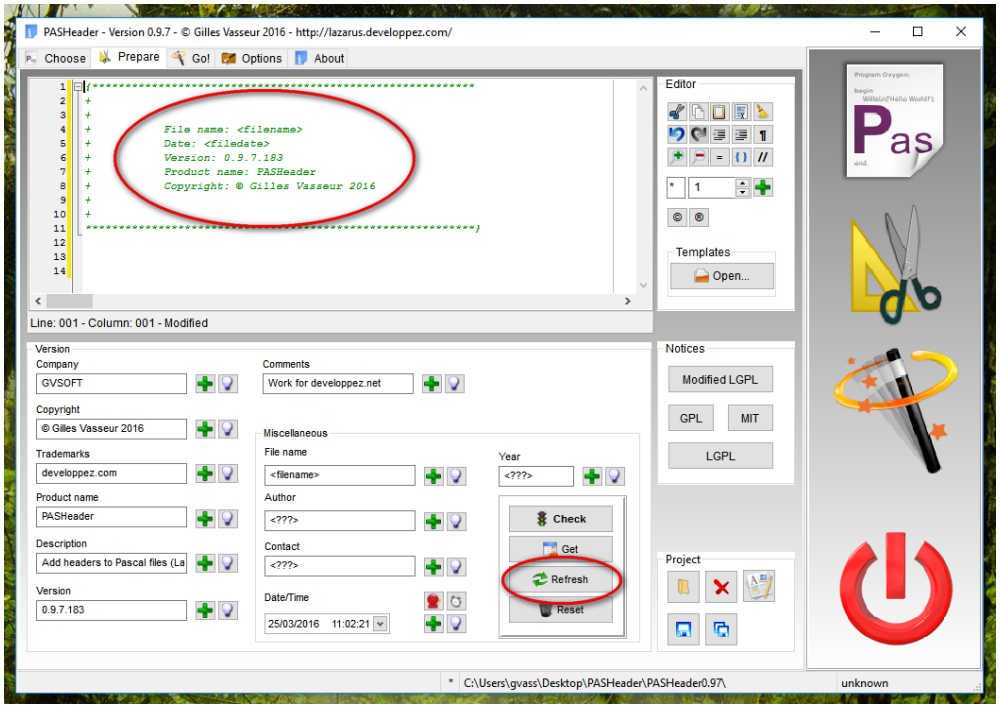


- Click "*get*" and observe the changes:

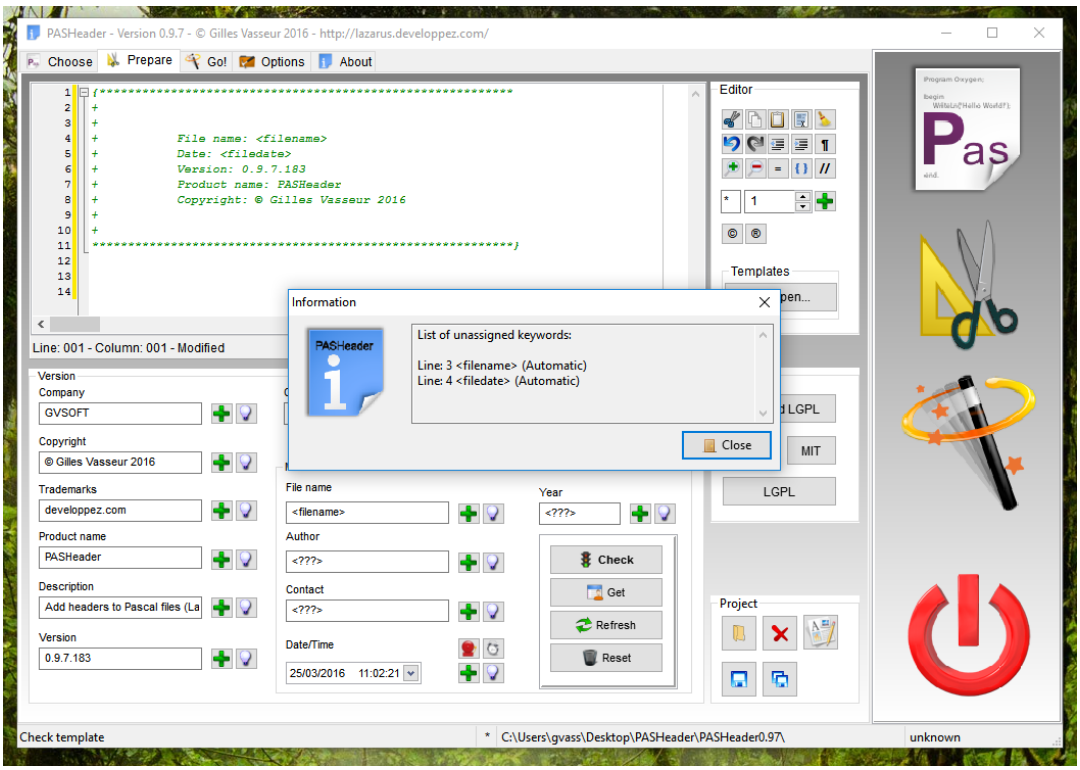


All fields included in the editor that can be extracted have been completed: source code in fact includes this type of information in the file "*pasheader.lpi*".

- Click "*refresh*" to request **PASHeader** to substitute the new data to the keywords of the template:



- Check that the header is correct by clicking "*check*":

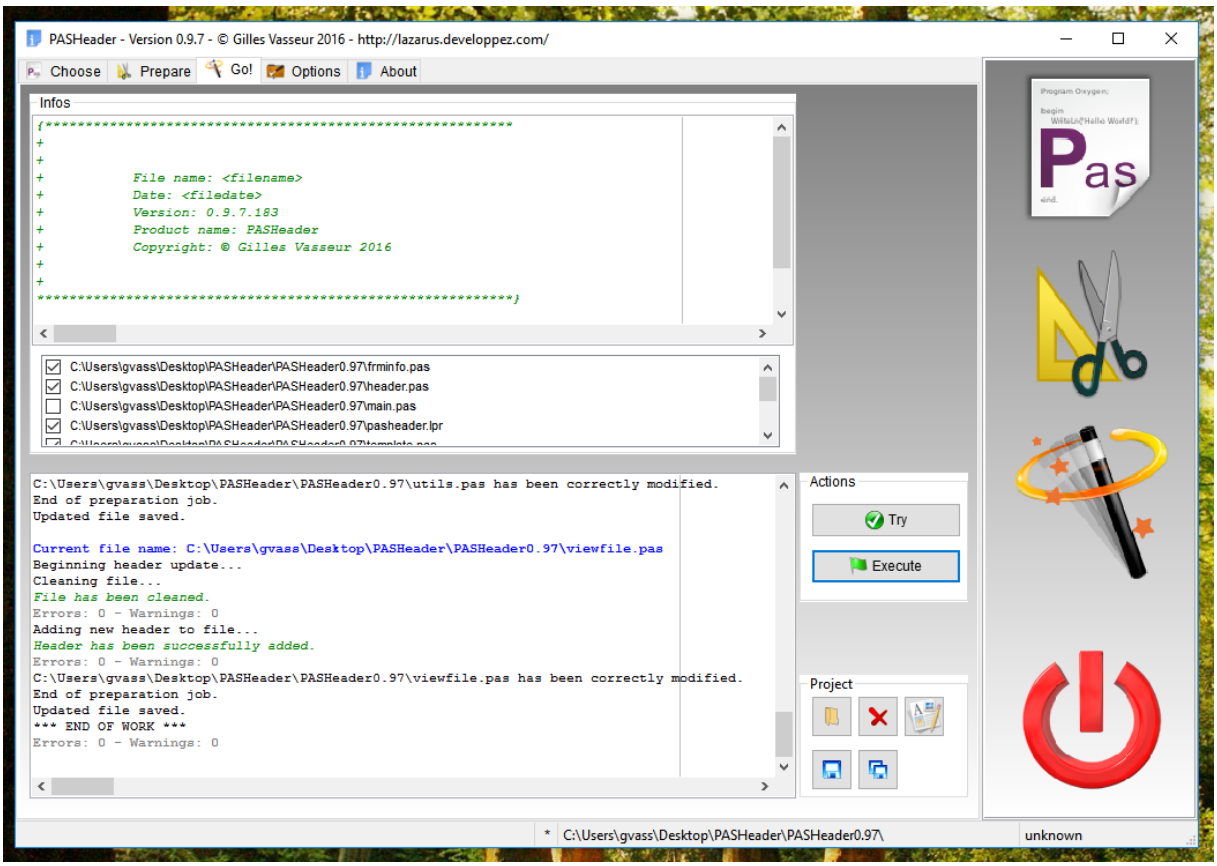


The new header is ready since there are only automatic fields that **PASHeader** will solve at runtime: <*filename*> and <*filedate*>.

How to apply the new header

On the execution page "*Go*!" you can run a simulation with the button" *try*" before changing the actual headers with the "*execute*" button.

- Click "*execute*":



By right clicking on their name followed by option "*view file*", you can now verify that the headers have been changed:

- checked files have been cleaned, the new header replacing the former;
- the unchecked file includes the new header placed before the former.

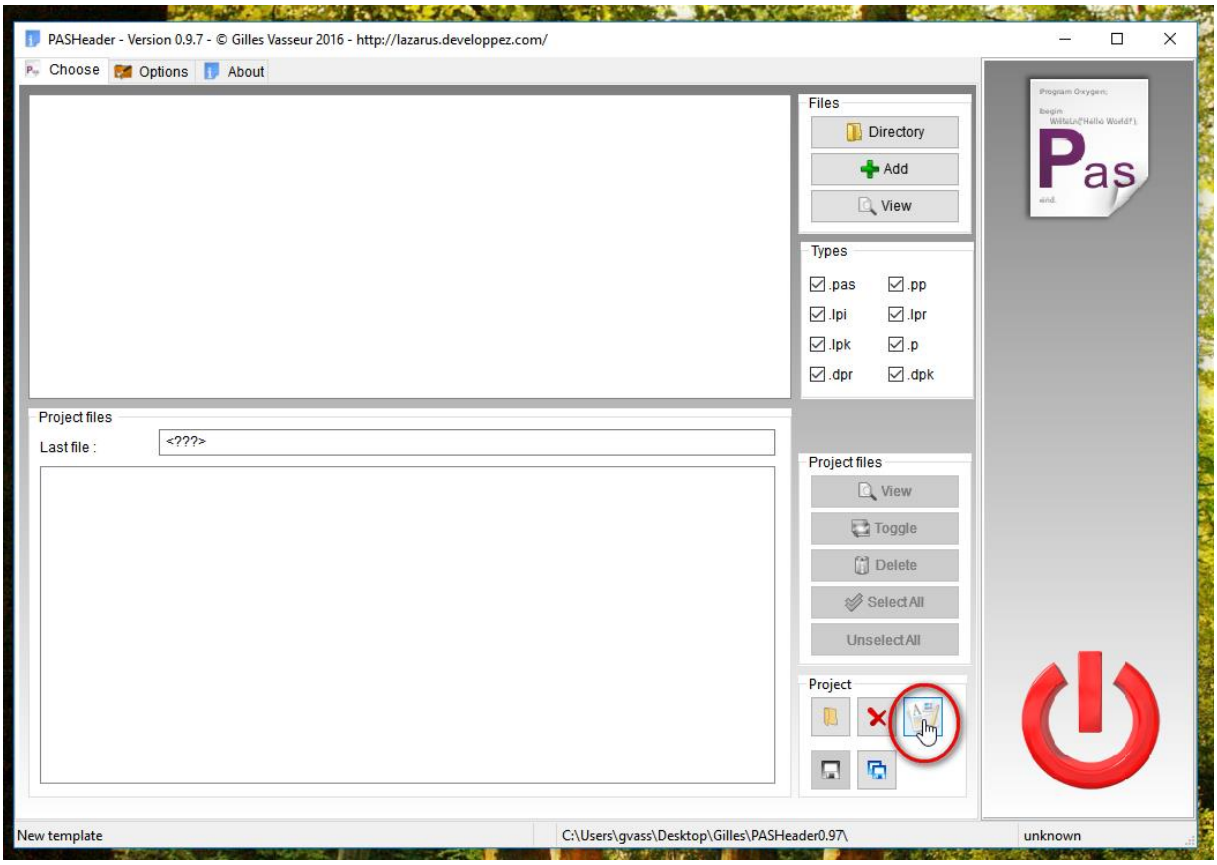
As variants of this tutorial, you can remove files from the list of selected files to prevent their headers from any changes. You can also extract data before loading the template and see that the fields are automatically updated when loading, without having to click "*refresh*". Finally, you can work with Delphi projects or Lazarus and Delphi packages. The version information will not be included with Delphi packages, but properly filled with Lazarus packages and Delphi projects.

How to create a new template

In the last tutorial, you will learn to create, modify and save a template header. As before, this tutorial assumes you have a basic knowledge of **PASHeader** and in particular that you are aware of its operating principles.

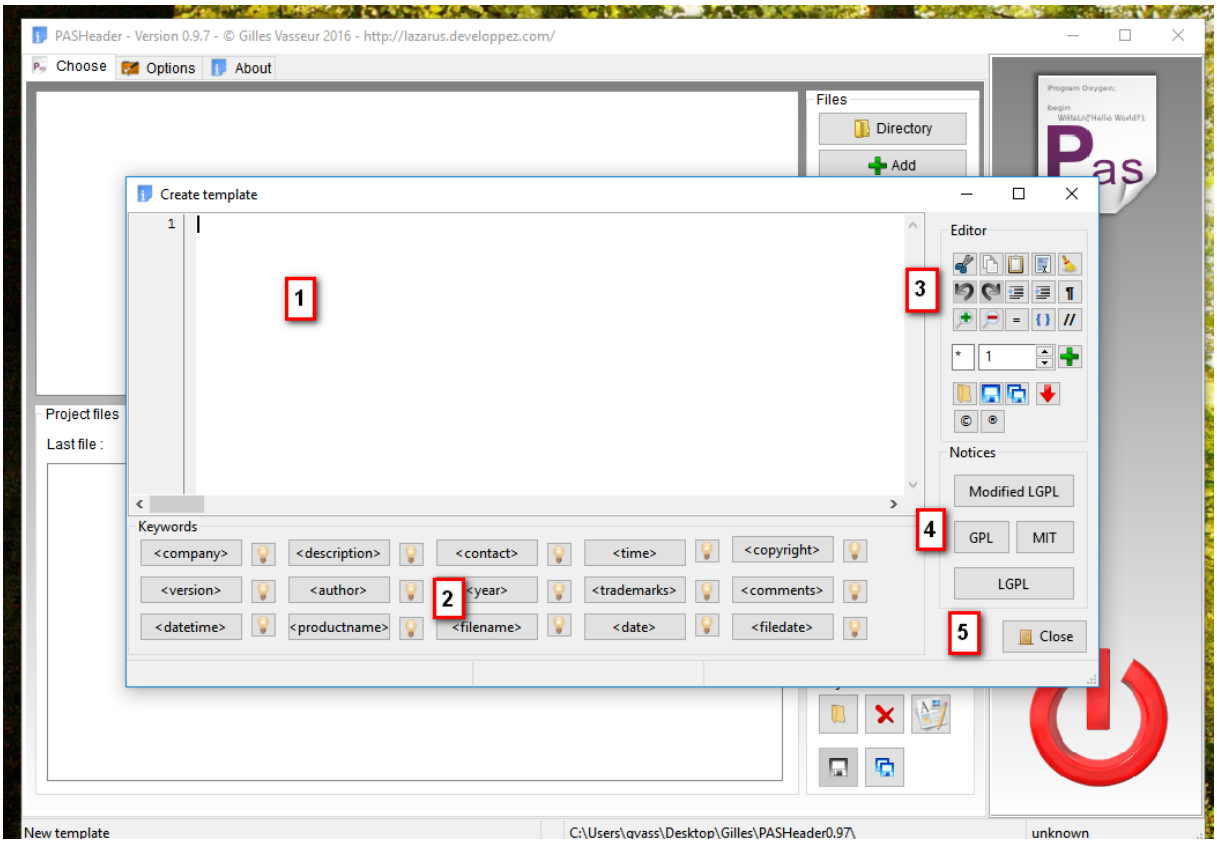
Procedure and features presented further will be little different from those already used to directly create a header. The specifics of editing templates mainly reside in the extensive use of keywords and saving or loading which will take into account the models and not specific projects.

- From the home screen, activate the template editor by clicking the appropriate button:



This button is always accessible, regardless of the level of advancement of the **PASHeader** project.

Here is the new window:



In 1, you will recognize an edit box for the template being processed. At first click on this area, the local taskbar indicates the cursor position and the changed status or not of the editor.

In 2 is a series of buttons labeled with a keyword, themselves accompanied by a button with an image of bulb. Those with text allow you to insert a keyword in the editor at the cursor location while those with picture are shortcuts to insert a standard text followed by the keyword.

In 3 is the panel of commands related to the editor. It looks very similar to the one already used for the header editor, apart from its lower part which introduces specific commands for templates.

In 4, you find the standard copyright notices that may be incorporated into the template at the cursor location.

In 5, the "*close*" button allows you to exit the editor. A confirmation will be made in case of modification without backup template.

As for editing headers, it is quite possible to work without these tools by manually creating a template.

Here is the table of keywords included in current version of **PASHeader**:

Keyword	Meaning	Status	Version information
<company>	Owner ot the product	Manual	Yes
<version>	Version of the product	Manual	Yes
<datetime>	Time and date of the product	Manual	No
<description>	Description of the product	Manual	Yes
<author>	Author of the product	Manual	No (except Lazarus project)
<productname>	Name of the product	Manual	Yes
<contact>	Contact (phone, email, address…)	Manual	Yes
<year>	Year of product	Manual	No
<filename>	File name	Automatic	No
<time>	Time of the product	Manual	No
<trademarks>	Trademarks	Manual	Yes
<date>	Date of the product	Manual	No
<copyright>	Copyright of the product	Manual	Yes
<comments>	Comments	Manual	Yes
<filedate>	Date of file	Automatic	No

Here is the control panel:

cut selected text to the clipboard;

copy selected text to the clipboard;

paste selected text from the clipboard;

select all;

clear the editor;

undo last command;

redo;

indent selected lines ;

unindent selected lines ;

¶ add a line break ;

increase the size of the text font;

reduce the size of the text font;

= make its original size to the text font;

{ } place the selected text in braces to a comment;

// place the comment sign at the beginning of the current line without moving the cursor;

reproduce the specific character in the first text box as many times as specified in the following areas;

load a template from media;

save the current template;

save the current template with a new name;

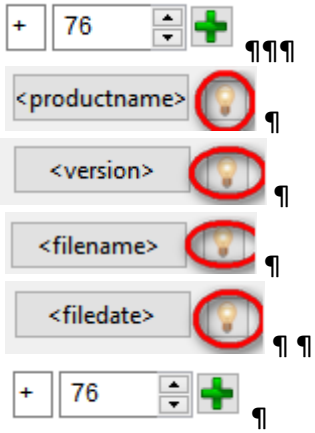
get the template from the header editor;

© insert the © character at the cursor location;

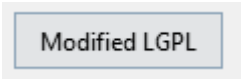
® insert the ® character at the cursor location.

Designing a template

To show you how shortcut keys make it easy to create a template, follow these steps:

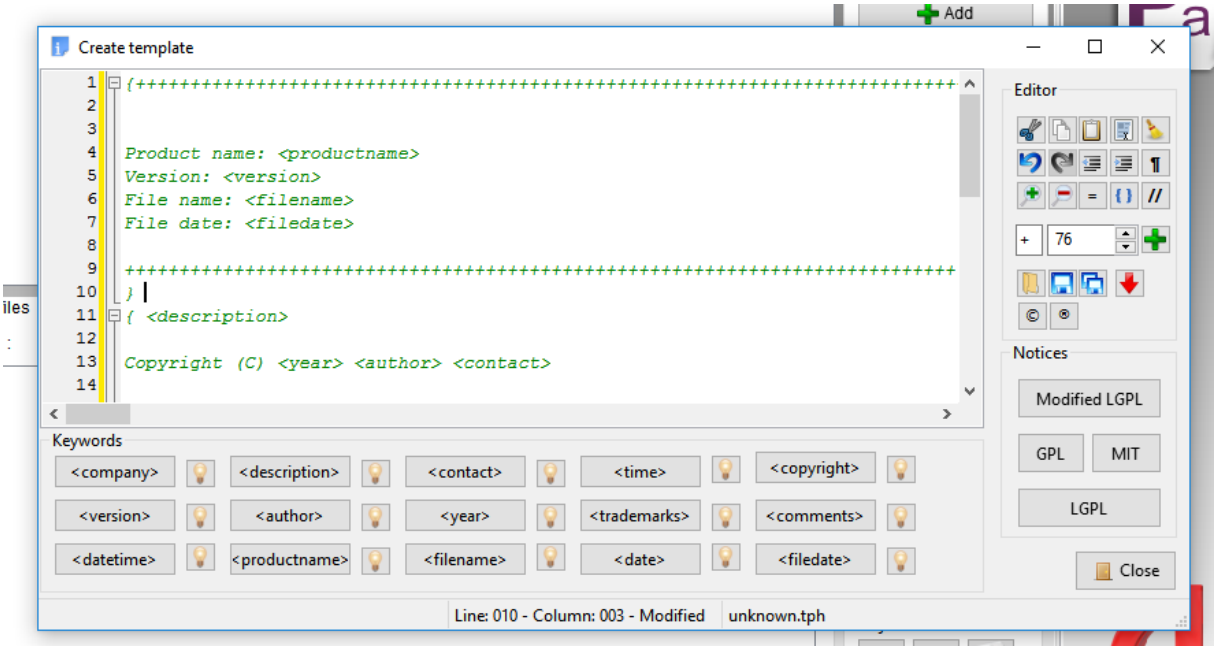


(select all)
{ } (braces around selected text)
¶ ¶



(copyright notice)

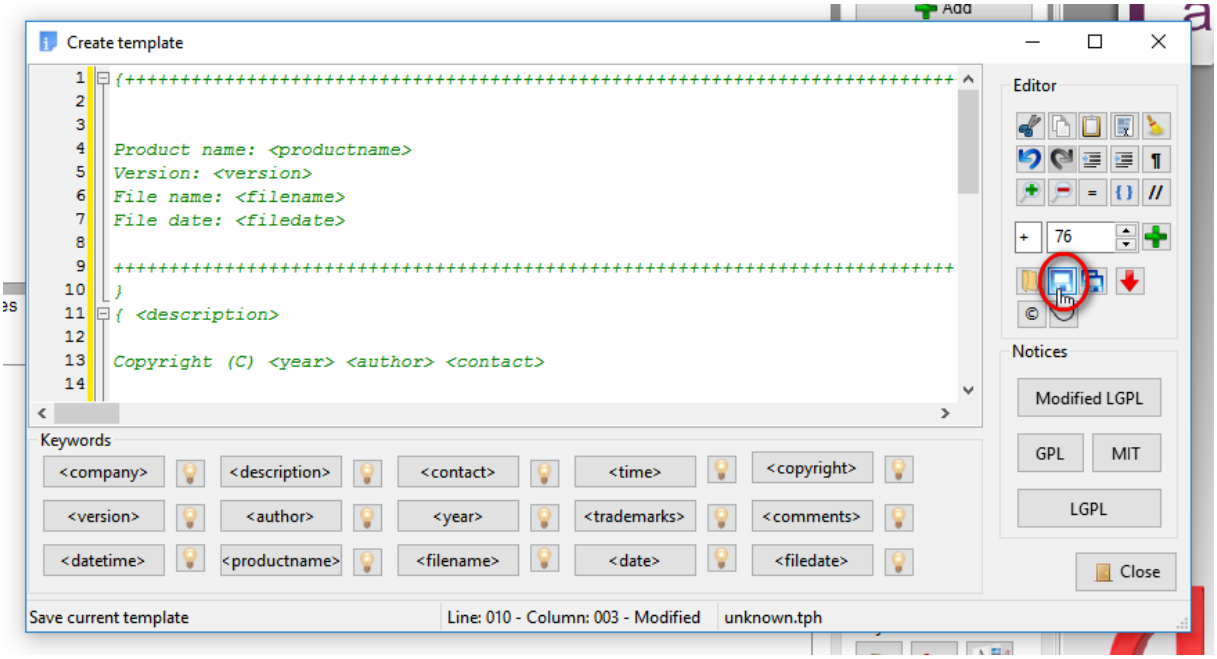
Without any keystroke, here's a preview of the resulting template:



You can adjust this model so that it corresponds exactly to your needs, **e.g.** indenting the central lines maybe too close to the left edge of the editor.

Saving your template

- Click the "*save*" button:



Save your model under the name that suits you, such as "*myfirstheader*".

Your template is now available for future headers that will brighten your files, while making them more informative at first glance!

Release notes

Version 0.96 – Possibility to load a project or a template from the explorer by dragging and dropping a file on **PASHeader**.

Version 0.97 – Version information recoverable from a "*.dproj*" file (Delphi project).

Table of contents